SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION

1b. RESTRICTIVE MARKINGS

3. DISTRIBUTION/AVAILABILITY OF REPORT
Unlimited

5. MONITORING ORGANIZATION REPORT NUMBER(S)
AFOSR-TR- 90-0006

6a. NAME OF PERFORMING ORGANIZATION
University of Minnesota

6b. OFFICE SYMBOL (If applicable)

7a. NAME OF MONITORING ORGANIZATION
AFOSR

6c. ADDRESS (City, State, and ZIP Code)
4-192 EE/CSci
200 Union Street SE
Minneapolis, MN 55455

7b. ADDRESS (City, State, and ZIP Code)
Bolling AFB, Washington DC 20332-6448

8a. NAME OF FUNDING/SPONSORING ORGANIZATION
AFOSR

8b. OFFICE SYMBOL (If applicable)
NM

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
AFOSR-87-0168

8c. ADDRESS (City, State, and ZIP Code)
Bldg. 410
Bolling AFB, DC 20332-6448

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO	WORK UNIT ACCESSION NO.
61102F	2304	~~[illegible]~~ A7	

11. TITLE (Include Security Classification)
Final Report - Structure From Motion

12. PERSONAL AUTHOR(S)
William B. Thompson

13a. TYPE OF REPORT
Final

13b. TIME COVERED
FROM 4/1/88 TO 9/30/88

14. DATE OF REPORT (Year, Month, Day)
1988 November 17

15. PAGE COUNT

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
Image Understanding, Time-Varying Image Analysis, Visual Motion, Optical Flow, Segmentation, [illegible] Intelligence, Space [illegible]

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Analysis of surface boundaries has been extended to situations in which a camera is able to actively track environmental surface points. Two problems were examined – the determination of relative depth at a boundary and the determination of the direction of motion. In both cases, the ability to actively track significantly decreases the complexity of the computations required. An analysis of the computational basis for the visual detection of moving objects has been completed. We have shown that moving object detection can exploit one or more of three general approaches. Each approach has particular strengths and weaknesses. Two significant results have been obtained in the area of motion-based segmentation. The first combines motion and contrast information in a boundary detection method that is both more reliable and more accurate than possible using only motion or only contrast. The integration is done in a manner involving little additional computation. Secondly, we have shown how motion information can be used to reduce ambiguity in the recognition of partially occluded objects. [illegible]

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION
Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL
Dr. Abraham Waksman

22b. TELEPHONE (Include Area Code)
(202) 767-5027

22c. OFFICE SYMBOL
NM

Previous editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

881228/same

FINAL REPORT - STRUCTURE FROM MOTION
AFOSR Contract AFOSR-87-0168

a. Objectives.

Our principal objective continues to be the development of robust computational approaches for estimating the spatial organization of a scene using time varying properties of image sequences. Three closely related problems are being pursued:

- *Active tracking of surface boundaries.*

  Much attention is currently being paid to problems involving *active vision*. An active vision system is able to at least partially control the manner in which perceptual information is acquired. Within the context of motion, several authors have argued that active tracking of moving objects or surface points provides additional constraints of use in solving structure-from-motion problems. We have shown that this is not in fact true. Active tracking can, however, significantly simplify some of the computations involved in analyzing visual motion.

- *Moving object detection.*

  The detection of moving objects is an important task for many robotics applications. With previous AFOSR support, we developed a series of algorithms for moving object detection in a variety of special situations. Under this contract, we have placed these methods under a coherent theoretical framework. As a result, it is now much easier to determine the difficulty of detection for a given situation and to apply the most appropriate detection method.

- *Motion-based segmentation.*

  We have done extensive research on methods for incorporating motion into the segmentation process. Motion-based segmentation is important because it provides more information than methods using only static cues. Two significant accomplishments have been achieved under this contract:

  - *Integrating motion and contrast for segmentation.*

    Motion-based edge detection is sensitive only to actual surface boundaries. As a result, ambiguity is reduced over methods based only on image contrast. Traditional brightness-based edge detection is far more precise at localizing edges, however. We have shown how edge detectors can be built that naturally incorporate the best aspects of brightness-based and motion-based edge detection.

  - *Occlusion-sensitive matching.*

    Our most important result under the current contract deals with improvements in object recognition that are possible using the results of our motion-based segmentation technique. Recognition in the presence of occlusion is difficult because it is hard to tell what features are part of the object being analyzed and what features are actually part of other objects partially occluding the object of interest. Our approach uses motion to differentiate between occluding and occluded surfaces. and then uses this information to remove irrelevant features from the classification process.

b. Status of research effort.

Active tracking of surface boundaries.

Others have argued that optical tracking of an environmental surface point significantly decreases the intrinsic complexity of various structure-from-motion problems. This is not in fact true. Tracking provides neither additional constraints nor other sorts of new information. This is easily seen by recognizing that all of the information in the tracking image is available in an image of the same scene without tracking. Tracking is accomplished by generating a rotation of the eye/camera system based on estimates of image drift such as optical flow at the image center. Once this rotational velocity is determined, a non-tracking image sequence can trivially be converted into the equivalent tracking sequence using standard techniques.

Active tracking can lead to important efficiencies in the implementation of certain structure-from-motion algorithms. We have developed two such methods:

- *Identification of occluding surface.*

  When a boundary element is visually tracked, the region to the side of the boundary corresponding to the occluding surface will have near-zero image flow. The region to the side of the boundary corresponding to the occluded surface will in general be associated with significant visual motion.

- *Determination of direction of observer motion.*

  When a boundary element is visually tracked, optical flow due to the more distant surface indicates the direction of observer motion. The flow vectors point in the direction of the image location corresponding to the line of sight coincident with the direction of translational motion. Multiple fixations over the field of view can be used to solve for the actual direction of translation.

The first of these techniques requires only the detection of regions with significant image motion, a far easier tasks than the comparisons required by previously known methods. The second technique eliminates difficulties due to camera rotation that plague most other solutions to this problem Additional discussion is presented in [5].

Moving object detection.

The reliable detection of moving objects is essential for many robotics applications. If the camera is stationary and illumination constant, this can be done by simple techniques which compare successive image frames, looking for significant differences. If the camera is moving, however, the problem is considerably more difficult. For a moving camera, both moving objects and stationary portions of the scene may be changing position with respect to the camera and thus generating visual motion in the imagery. A moving camera leads to difficulties because of the need to determine objects moving with respect to the environment, rather than the much easier problem of finding objects moving with respect to the camera. General solutions based only on vision are computationally complex and likely to be numerically unstable. If partial information is available about camera motion and/or scene structure, however, robust motion detection methods are possible.

We have shown that possible approaches to this problem fall into three categories:

- *Violations of motion epipolar constraint.*

  Translational motion produces a flow field radially expanding from a "focus of expansion" (FOE). Any flow vectors violating this constraint are due to moving objects.

- *Comparison of optical flow and other depth information.*

  While patterns of optical do not uniquely specify depth, they do constrain the possible values. Motion-based constraints on possible depth can be combined with static constraints obtained from cues such as stereo. Violations of the combined constraints indicate that moving objects are present.

- *Violations of rigid object constraint.*

  Only certain patterns of optical flow can correspond the the imagery produced by a moving, rigid, three-dimensional object. While we have not yet researched this approach extensively, there is reason to believe that it may be possible to determine whether or not this rigidity constraint is actually satisfied. If so, distinctly non-rigid motion corresponds to moving objects.

Understanding the theoretical underpinnings of moving object detection has several advantages. Perhaps most importantly, it is now possible to determine under what situations a particular approach will work without having to examine the details of a specific algorithm. Likewise, the strengths and weaknesses of whole classes of algorithms can be investigated at one time. Finally, we expect that better performing algorithms will arise from a more complete understanding of the basic constraints involved in the problem. More information can be found in [1].

Motion-based segmentation.

Edge detection algorithms based on visual motion perform significantly differently than those based on brightness. Previous attempts to combine motion and contrast information in edge detection have not recognized these differences. Static cues such as contrast edges give good spatial localization, but are subject to highly ambiguous interpretations. Visual motion is a robust indicator of surface boundaries, but does not yield precise information on the location of the boundary. The approach described in [4] accurately locates edges due to surface boundaries, without generating many "false" edges. Furthermore, the combined method adds minimal computational complexity to the edge detection process.

Our most important result under the current contract deal with the problem of recognizing partially occluded objects. Most existing matching algorithms that are tolerant of occlusion look for a partial correspondence between model and image features. If a partial match is found, unmatched model components are assumed to be hidden by an occlusion. This approach leads to difficulties because of the chances for partial matches occurring coincidentally. In our method, motion-based information about occlusion boundaries is used to explicitly identify model features that will not be visible in the image. Most of the remaining model features should be findable if the match is in fact correct. Occluded model features are determined based directly on image properties at boundaries, rather than just on the absence of an image feature at some expected location. The result is a significant decrease in ambiguity. Details are found in [4].

c. Publications.

[1] W.B. Thompson and T.C. Pong, "Detecting Moving Objects," submitted to *International Journal of Computer Vision.*

[2] W.B. Thompson and T.C. Pong, "Detecting Moving Objects," *Proceedings of the First International Conference on Computer Vision,* June 1987.

[3] W.B. Thompson, L.G. Craton, and A. Yonas, "The $2\frac{3}{4}$-D Sketch," *Proceedings of the AAAI Workshop on Physical and Biological Approaches to Computational Vision,* March 1988.

[4] W.B. Thompson and R.P. Whillock, "Occlusion-Sensitive Matching," *Proceedings of the Second International Conference on Computer Vision,* December 1988.

[5] W.B. Thompson, "Structure-From-Motion By Tracking Occlusion Boundaries," *Proceedings IEEE Workshop on Visual Motion,* March 1989.

d. Scientific Collaborators.

Research Assistants:

King Chu
Martin Kenner
Steven Savitt
Elizabeth Stuck
Rand Whillock

Collaborating Faculty:

Herbert Pick
Ting-Chuen Pong
Albert Yonas

Detecting Moving Objects

William B. Thompson
Ting-Chuen Pong

Computer Science Department
University of Minnesota
Minneapolis, MN 55455

Submitted to the *International Journal of Computer Vision.*

Abstract

The detection of moving objects is important in many tasks. This paper examines moving object detection based primarily on optical flow. We conclude that in realistic situations, detection using visual information alone is quite difficult, particularly when the camera may also moving. The availability of additional information about camera motion and/or scene structure greatly simplifies the problem. Two general classes of techniques are examined. The first is based around the motion epipolar constraint – translational motion produces a flow field radially expanding from a "focus of expansion" (FOE). The second class of methods is based on comparing observed optical flow with other information about depth. Examples of several of these techniques are presented.

1 Introduction.

One important function of a vision system is to recognize the presence of moving objects in a scene. If the camera is stationary and illumination constant, this can be done by simple techniques which compare successive image frames, looking for significant differences. If the camera is moving, the problem is considerably more complex. For the purposes of this discussion, *moving objects* are taken to be any objects moving with respect to the stationary portions of the scene, which we refer to as the *environment*. For a moving camera, both moving objects and stationary portions of the scene may be changing position with respect to the camera and thus generating visual motion in the imagery. A moving camera leads to difficulties because of the need to determine objects moving with respect to the environment, rather than the much easier problem of finding objects moving

This work was supported by AFOSR contract AFOSR-87-0168 and NSF Grants DCR-8500899 and IRI-8722576.

A preliminary version of this paper appeared in *The Proceedings of the First International Conference on Computer Vision*, London, June 1987.

with respect to the camera. In this paper, we deal with the problem of detecting moving objects from a moving camera based on optical flow.

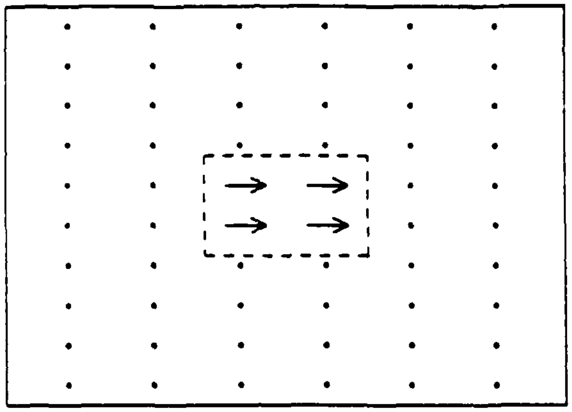


Figure 1: Is The Central Region a Moving Object?

The visual detection of moving objects is a surprisingly difficult task. A simple example illustrates just how serious the problem can be. Consider the optical flow field shown in figure 1 which appears to show a small, square region in the center of the image moving to the right and surrounded by an apparently stationary background. Such a flow field can arise from several equally plausible situations: 1) The camera is stationary with respect to the environment, and the central region corresponds to an object moving to the right. 2) The camera is moving to the left with respect to the environment, most of the environment is sufficiently distant so that the generated optical flow is effectively zero, while the central region corresponds to a surface near to the camera but stationary with respect to the environment. 3) The camera and object are moving with respect to both the environment and each other, though the environment is sufficiently distant so that there is no perceived optical flow. It is not possible to tell whether or not this seemingly simple pattern corresponds to a moving object![1]

Figure 1 provides one example of why a general and reliable solution to the problem of moving object detection based only on optical flow is not feasible. Robust solutions require that additional information about camera motion and/or scene structure be available. In this paper, we examine a variety of types of information that might be available. Each information source places constraints on the optical flow fields that can be generated by a camera moving through an otherwise static environment. Violations of these constraints are thus necessarily due to moving objects.

[1] The flow pattern in figure 1 provides little information about actual camera motion. Apparently stationary image regions can be due to the viewing of distant surfaces and/or rotational motion that tracks a surface point, keeping it at a fixed point in the field of view. Even with significant non-zero flow existing over the whole of the image, ambiguities exist between flow patterns due to translational motion and due to rotational motion [1].

2 Background.

An extensive literature has developed on computational approaches to the analysis of visual motion (e.g., see [2]). The majority of this work deals with what Ullman [3] has called the structure-from-motion and motion-from-structure problems. Visual motion is used to determine the three-dimensional position of surface points under view and/or the parameters of motion relating camera and object. Almost without exception, papers describing structure-from-motion and motion-from-structure algorithms deal only with a single, rigid object in the field of view. If the problem of separately moving objects is mentioned at all, it is in a comment that the image must be segmented into separately moving objects before the method being described is applied.

Some work has been done on the segmentation of images based on visual motion. The easiest form of this problem occurs with a camera known to be stationary. In such circumstances, object motion leads to significant temporal differences in an image sequence. Such differences correspond to moving objects, and furthermore can be used to estimate the boundaries of the objects (e.g., [4, 5]). More classical edge-detection techniques can also be applied to time-varying imagery [6, 7, 8, 9, 10, 11]. Such approaches work for both moving and stationary cameras. When the camera is moving, however, sharp spatial changes in visual motion can correspond to either the boundaries of moving objects or to depth discontinuities between two rigidly attached surfaces. As a result, motion-based edge detection is not sufficient to detect moving objects.

Jain is one of the few researchers to deal directly with the problem of detecting moving objects using a moving camera [8]. His approach exploits the motion epipolar constraint which says that for translational camera motion with respect to a static environment, optical flow will expand radially from a focus of expansion corresponding to the direction of translation. For translational motion, any flow values violating the epipolar constraint must be due to moving objects in the scene. Unfortunately, this approach requires knowledge of the direction of translation and does not work if the motion has a rotational component.

3 Possible approaches.

At least three general approaches to moving object detection are possible. Each exploits a particular constraint that must hold if a camera is moving through an otherwise static environment. Detecting moving objects becomes equivalent to a search for violated constraints.

- *Motion epipolar constraint.*

  Translational camera motion produces a distinctive optical flow pattern. Flow vectors appear to radiate out from a "focus of expansion" (FOE) corresponding to the line of sight coincident with the direction of motion. This has the effect of constraining the orientation of flow vectors. Visual motion which violates this orientational constraint must be due to moving objects. Under some circumstances, the motion epipolar constraint may still be used when camera rotation is added to the translational movement.

- *Depth/flow constraint.*

  The optical flow generated by a surface point is a function of the relative motion between camera and surface and of the range to the surface. If range values are available. then inconsistencies between optical flow. range, and observer motion signal moving objects.

- *Rigidity constraint.*

  A scene containing moving objects can be thought of as undergoing non-rigid motion with respect to the camera. Structure-from-motion techniques which are sensitive to the presence of non-rigid motion can thus be used to detect moving objects.

This paper will concentrate on epipolar and depth/flow methods. Though potentially effective, methods based directly on the rigidity constraint require longer frame sequences. temporal derivatives of optical flow. and/or a wide field of view to enhance perspective effects.

4 Presumptions.

Many theoretically plausible techniques for analyzing visual motion are ineffective in practice. Typically, the assumptions on which these techniques are either explicitly or implicitly founded do not accurately represent real problems. For this work, we start with the presumption that motion detection algorithms should be designed with the following properties in mind:

- *The field of view may be relatively narrow.*

  Motion detection should not depend on the use of wide angle imaging systems. Such systems may not be available in a particular situation, and if used may increase the difficulty or recognizing small moving objects. As a result, detection algorithms should not depend on subtle properties of perspective.

- *The image of moving objects may be small with respect to the field of view.*

  This is clearly desirable for reliability. Moving objects may be far away and subtended by relatively small visual angles. We need methods capable of identifying single image points, or at least small collections of points, as corresponding to moving objects. Detection algorithms thus cannot depend on variations in flow over a potentially moving object.

- *Estimated optical flow fields will be noisy.*

  No method is capable of estimating optical flow with arbitrary accuracy. Motion detection based on optical flow must be tolerant of noisy input.

5 The Optical Flow Equation.

The basic mathematics governing the optical flow generated by a moving camera is well known. Our notation is similar to [12], using a coordinate system fixed to the camera (e.g., the world can be thought of as moving by a stationary camera). Optical flow values are a function of image location, the relative motion between the camera and the surface point corresponding to the image location, and the distance from the camera to the corresponding surface point:

$$\mathbf{F}(\mathbf{p}) = \frac{\mathbf{F}_t(\mathbf{p})}{r(\mathbf{p})} + \mathbf{F}_r(\mathbf{p}) \tag{1}$$

$$\mathbf{F}_t = (-U + xW \ , \ -V + yW) \tag{2}$$

$$\mathbf{F}_r = \Big(Axy - B(x^2 + 1) + Cy \ , \ A(y^2 + 1) - Bxy - Cx\Big) \tag{3}$$

where $\mathbf{F}$ is the optical flow at image location $\mathbf{p} = (x, y)$, x and y are normalized by the focal length, $r(\mathbf{p})$ is the range from the camera to the surface point imaged at $\mathbf{p}$, $\mathbf{T} = (U, V, W)^T$ specifies the translational velocity of the camera, and $\boldsymbol{\omega} = (A, B, C)^T$ specifies camera rotation.

Most work on the analysis of optical flow has dealt with a camera moving through an otherwise static environment or, equivalently, a single rigid object moving in front of a fixed camera. In such cases, single values of $\mathbf{T}$ and $\boldsymbol{\omega}$ govern the flow over the whole image. If moving objects are present, then the relative motion between camera and environment will be different than the relative motion between camera and moving object. Notationally, we will specify the camera motion with respect to the environment by $\mathbf{T}^{(env)}$ and $\boldsymbol{\omega}^{(env)}$. The parameters specifying the relative motion between the camera and an arbitrary scene point $\mathbf{p}$ will be indicated by $\mathbf{T}^{(\mathbf{p})}$ and $\boldsymbol{\omega}^{(\mathbf{p})}$. $\mathbf{p}$ lies on a moving object if $\mathbf{T}^{(\mathbf{p})} \neq \mathbf{T}^{(env)}$ and/or $\boldsymbol{\omega}^{(\mathbf{p})} \neq \boldsymbol{\omega}^{(env)}$.

6 Detection based on Epipolar Constraint.

If complete information about instantaneous camera motion is available, then $\mathbf{T}^{(env)}$ and $\boldsymbol{\omega}^{(env)}$ are known. If the camera is translating but not rotating with respect to the background, $\boldsymbol{\omega}^{(env)} = 0$, $\mathbf{F}_r = 0$, and all flow vectors due to the moving image of the background will radiate away from a *focus of expansion* (FOE). From equations 1 and 2, it is easy to see that the image plane location of the FOE is at:

$$(x, y)_{foe} = \left(\frac{U}{W}, \frac{V}{W}\right) \tag{4}$$

While the location of the FOE depends only on the direction of translation and not on the speed, it is important for detectability that the speed be sufficient to generate measurable optical flow. The FOE is not restricted to lie within the visible portion of the image (and in fact may be a focus of contraction). An FOE at ∞ corresponds to pure lateral motion, which generates a parallel optical flow pattern.

6.1 Direct use of motion epipolar constraint.

For pure translational motion. the direction of motion specifies the direction of optical flow associated with any surface point stationary with respect to the environment:

$$\theta_{foe} = tan^{-1} \frac{V - Wy}{U - Wx} \tag{5}$$

where θ_{foe} is the expected flow orientation at the point (x, y). predicted using the motion epipolar constraint. Note that this equation is still well defined when $W = 0$. corresponding to a focus of expansion at ∞ in image coordinates. Any flow values with a significantly different direction correspond to moving objects [3]. (The converse is not necessarily true. It is possible that moving objects coincidentally generate flow values compatible with this constraint.) This approach requires the estimation of only the direction of flow, not either the magnitude or spatial variation of flow.

Camera rotation introduces considerable complexity. Knowledge of camera motion no longer constrains the direction of background flow. Nevertheless, at a given point **p**, flow is constrainted to a one-dimensional family of possible vector values. The family is given by (1–3) where r ranges over all positive values. The analysis can be simplified because of the linear nature of (1). $\mathbf{F}_r$ depends only on the parameters of rotation and not on any shape property of the environment. Because the value of $\mathbf{F}_r$ at a particular point **p** does not depend on $r(\mathbf{p})$, it can be predicted knowing only ω. At every point within the field of view, this value can be subtracted from the observed optical flow field, leaving a *translational flow field*:

$$\mathbf{F}_{trans} = \mathbf{F} - \mathbf{F}_r \tag{6}$$

This field behaves just as if no rotation was occurring, and thus moving objects can be located using the FOE technique described above. For the remainder of this paper, when rotation is present, we will take the term FOE to refer to the focus of expansion of this translational field.

In principle, even if camera motion is not known $\mathbf{T}^{(env)}$ and $\omega^{(env)}$ may be estimated from the imagery (e.g., [12]), subject to a positive, multiplicative scale factor for $\mathbf{T}^{(env)}$. Two serious problems exist, however. Narrow angles of view make estimation of camera motion difficult, as significantly different parameters of motion and surface shape can yield nearly identical optical flow patters [1]. In addition, techniques such as [12] uses a global minimization approach which will not perform well if moving objects make up a substantial portion of the field of view. A clustering approach (e.g., [13]) can be made tolerant of the moving objects, though great difficulty can be expected dealing with a five dimensional cluster space.

6.2 Indirect use of motion epipolar constraint.

The motion epipolar constraint has an important implication for motion analysis methods that operate only over small image neighborhoods. Away from the FOE, $\mathbf{F}_t(\mathbf{p})$ and $\mathbf{F}_r(\mathbf{p})$ vary slowly with **p** (equations 2 and 3). Over a small neighborhood, both $\mathbf{F}_t(\mathbf{p})$ and $\mathbf{F}_r(\mathbf{p})$ are essentially constant. As a result, over a small neighborhood, the component of flow due to rotational motion is essentially constant, while the translational flow, $\mathbf{F}_{trans}$, varies only by a scalar multiple dependent on depth. That is, over the neighborhood $\mathbf{F}_{trans}$ is essentially constant in direction. We can use this result to simplify problems arising from rotational camera motion. In one technique, we explicitly compensate for rotation. In a second technique, active tracking of potentially moving objects leads to a particularly simple computational scheme.

6.2.1 Known rotation.

Often, information about camera rotation is available, even when the direction of translation is not known. Non-visual information about camera motion often comes from inertial sources. Such sources are much more accurate in determining rotation than translation. Rotation involves a continuous acceleration which is easily measured. The determination of translation requires the integration of accelerations, along with a starting boundary value. Errors in estimated translation values rapidly accumulate. A simple technique allows the detection of moving objects when only camera rotation is known.

If all motion parameters are known, knowledge of camera rotation makes it possible to compute the translational flow field, $\mathbf{F}_{trans}$. Knowledge of translation can then used to locate the FOE and thus constraint the direction of flow vectors associated with the environment. If only rotation is known, it is still possible to determine the translational flow field, but not the FOE. Visual methods an be applied to the translational flow field to estimate the location of the FOE, but these methods suffer from a number of practical limitations when applied to noisy data.

An alternate approach can be used which does not require the prior determination of the FOE. The translational flow field extends radially from the focus of expansion. From the arguments given above, we know that over any local area away from the FOE, variations in the *direction* (but not necessarily magnitude) of the translational flow field will be small. Flow arising due to moving objects is of course not subject to this restriction. The gradient of flow field direction can thus be used to detect the boundaries of moving objects. At these boundaries, flow direction will vary discontinuously[2]

A complementary technique is available to deal with situations in which translation but not rotation is known. We can expect these situations to be rare, however. If the direction of translation were known over some interval of time, it would be an easy matter to determine the rotation by examining the rate of change of direction.

[2] Marr [14] claims "if direction of [visual] motion is ever discontinuous at more than one point – along a line, for example, – then an object boundary is present." Note that this is only necessarily true if no camera rotation is occurring (or equivalently, if camera rotation has been normalized by using the translational flow field).

6.2.2 Active tracking.

A vision system which can actively control camera direction is capable of tracking regions of interest over time. keeping some particular object centered within the field of view. Tracking regions of interest is desirable for many reasons other than the detection of moving objects (e.g., [15]), though the analysis of imagery arising from a tracking camera has not received much study by the computer vision community. If there are significant variations in depth over the visible portion of the background and if moving objects are relatively small with respect to the field of view, then moving object detection based on tracking can be accomplished without any actual knowledge of camera motion. (For motion detection, the tracking can easily be simulated if the camera is not actively controllable.)

If an object is being tracked. then its optical flow is zero.[3] Flow based methods for determining whether or not a tracked object is moving must depend wholly on the patterns of flow in the background. Object tracking helps in moving object detection because it minimizes many of the difficulties due to camera rotation. When dealing with instantaneous flow fields, we can decompose the problem by considering all translational motion to be due to movement of the camera platform and all rotational motion due to pan and tilt of the camera to accomplish the tracking. (We will disregard any effects due to spin around the line of sight.) Consider the effect of tracking a point that is in fact part of the environment. Tracking is effected by generating a rotational motion that exactly compensates for the translational flow at the center of the image. This is accomplished by choosing Fr such that:

$$\mathbf{F}_r(0,0) = -\frac{\mathbf{F}_t(0,0)}{r(0,0)} \tag{7}$$

For a small enough neighborhood, $\mathbf{F}_t$ and $\mathbf{F}_r$ can be treated as constant, leading to the following flow equation:

$$\mathbf{F}_{track}(\mathbf{p}) = \left(\frac{1}{r(\mathbf{p})} - \frac{1}{r(0,0)}\right)\mathbf{F}_t \tag{8}$$

The effect on the optical flow field is that in the neighborhood of the tracked point, the direction of flow will be approximately constant (modulo 180°), with a magnitude dependent on the difference between the range to the corresponding surface point and the range to the tracked point.

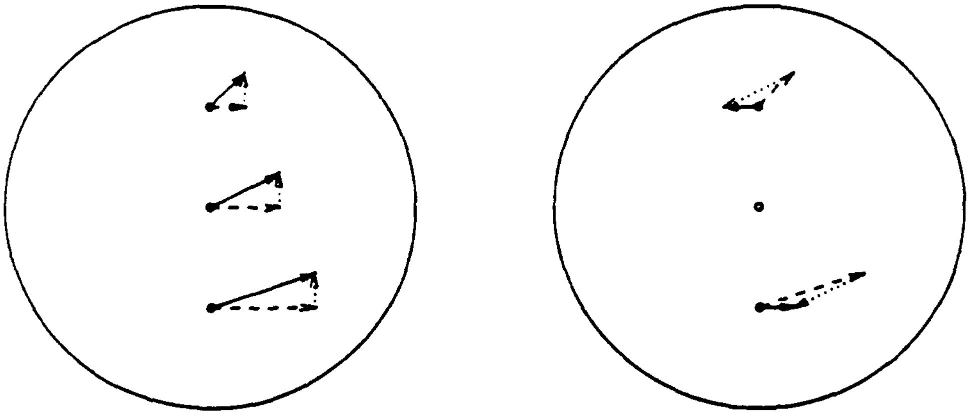
Now, consider tracking a point that is moving with respect to the environment. If environmental surface points are visible in the neighborhood of the tracked point, $\mathbf{F}_t$ and $\mathbf{F}_r$ are no longer constant within the neighborhood. For environmental points:

$$\mathbf{F}_{track}(\mathbf{p}) = \frac{\mathbf{F}_t^{(env)}}{r(\mathbf{p})} + \mathbf{F}_r^{(env)} - \frac{\mathbf{F}_t^{(object)}}{r(0,0)} \tag{9}$$

$\mathbf{F}_t^{(env)}$, $\mathbf{F}_r^{(env)}$, and $\mathbf{F}_t^{(object)}$ will in general differ in orientation. If there is a variation in range to visible environmental points, then there will be a variation in direction of observed flow over the neighborhood. (Note that detection is not possible if there is no variation in $r(\mathbf{p})$ over the visible environment. This situation is similar to that depicted in figure 1.)

[3] To simplify discussion, we ignore the case of an object rotating in depth. The method developed does in fact deal effectively with this situation.

Figures 2 and 3 illustrate the effect. Figure 2 shows the optical flow over a neighborhood in which no motion is occurring with respect to the environment. Figure 2a shows the flow before any tracking motions are initiated. The dashed line indicates the translational component of flow. The rotational component of flow is indicated by the dotted line. The solid line is the observed optical flow. the sum of the translational and rotational components. The translational components are parallel. The variations in magnitude correspond to underlying variations in range. The rotational components are constant over the neighborhood. Note that the observed flow varies in orientation - as previously indicated. orientational variability alone is not enough to detect moving objects. Figure 2b shows the flow that results when the point in the center of the region is being tracked. The center flow is of course zero. The dashed lines now indicate the flow that would be observed without tracking. The dotted lines indicate the rotational flow that is introduced to stablize the center point withing the field of view. The solid line shows the resulting optical flow. Note that the flow vectors are parallel, but in this case differ by 180°.



2a: Before tracking

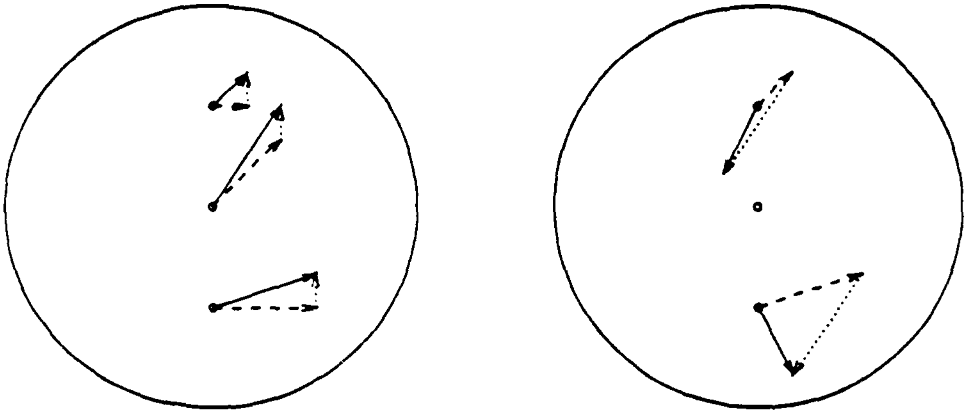
2b: After tracking

Figure 2: Tracking a Stationary Surface Point.

Figure 3 shows the same flow vectors in the case where the center point corresponds to a moving object and the two other points correspond to portions of the environment. Note that in figure 3a, the translational flow varies significantly in orientation. If we actually knew the translational flow, this fact would be enough to determine that a moving object was present. Without information about camera rotation, however, we must resort to more indirect methods.

7 Detection Based on Flow/Depth Constraint.

Recently, efforts have been made at developing integrated approaches to analyzing stereo and motion (e.g., [6, 16]). These approaches simultaneously deal with motion and stereo disparity, either by comparing flow fields taken from different viewing positions or by establishing point correspondences over both time and viewing directions. Similar multi-cue analysis can greatly aid in the detection of moving objects. We claim, however, that it is not necessary to adopt a



3a: Before tracking

3b: After tracking

Figure 3: Tracking a Moving Object.

strategy requiring the unified low-level integration of motion and stereo. Rather, depth estimates from whatever sources are available can be used. In addition to stereo, these sources can include the full range of non-motion depth cues: familiar size, focus, gradients of various properties, aerial perspective, and many more [17]. Furthermore, while precise estimates of depth are obviously useful, relative depth or coarse approximations to depth can also aid in the analysis.

7.1 Objects moving on surfaces.

Knowledge of the shape of environmental surfaces can be used to simplify the motion detection problem. Scene structure may be known precisely (e.g., the range to visible surface points) or in terms of general properities (e.g., significant depth discontinuities can be expected). If moving objects must remain in contact with environmental surfaces (e.g., vehicular motion), a less complex technique depending only on knowing the image plane locations corresponding to discontinuities in range is possible. If no objects are moving within the field of view, equations (1–3) show that flow varies inversely with distance for fixed p. Both $\mathbf{F}_r$ and $\mathbf{F}_t$ vary slowly (and continuously) with p. Discontinuities in $\mathbf{F}$ thus correspond to discontinuities in r. This relationship holds only for relative motion between the camera and a single, rigid structure. When multiple moving objects are present, equation 1 must be modified so that there is a separate $\mathbf{F}_r{}^{(i)}$ and $\mathbf{F}_t{}^{(i)}$ specifying the relative motion between the sensor and each rigid object. Discontinuities in flow can now arise either due to a discontinuity in range or due to the boundaries of a moving object. If independent information is available on the location of range discontinuities, and other discontinuities in flow must be due to moving objects.

The motion detection problem becomes particularly simple if the environment is planar. In this case, depth discontinuities are not possible and *any* discontinuity in flow (either direction or magnitude) corresponds to the boundary of a moving object. Note that it is not sufficient to know simply that the environment is a "smooth" surface. From some viewing positions, even smooth surfaces may exhibit range discontinuities.

7.2 Direct comparison of depth and flow.

A simple way of combining depth and visual motion to detect moving objects is possible if accurate 3-D position information is available for a sufficient number of surface points in the environment and on any moving objects. If both the optical flow and the depth are known for a collection of surface points in the environment, then equations (1)–(3) can be used to create a system of equations which can be solved for the parameters of motion $\mathbf{T}^{(env)}$ and $\omega^{(env)}$. (Knowing depth values makes this an easier task than the standard structure-from-motion problem.) If the collection of points includes some values associated with the environment and others associated with one or more objects moving with respect to the environment, the system of equations used to solve for $\mathbf{T}$ and ω will be inconsistent. Checking the system for consistency can therefore be used as a test for the presence of a moving object (e.g., a test for non-rigid motion in the field of view). Only the consistency of the system is important. The actual values of $\mathbf{T}$ and ω are not relevant to the detection problem.

7.3 Indirect comparison of depth and flow.

The availability of accurate 3-D position estimates depends in large part on having accurately calibrated camera systems. Not only is this calibration difficult, but it is continuously subject to variability due to mechanical compliance. *Relative* measures of visible motion and/or stereo can be used to avoid this calibration problem (e.g., [18]). For example, Reiger and Lawton have shown how to use local spatial differences to minimize difficulties due to rotation [19]. If no moving objects are visible, then large local differences in flow can only be due to a change in depth. If $\mathbf{p}^{(i)}$ and $\mathbf{p}^{(j)}$ are image points on either side of such a boundary, then from equation (1) we have:

$$\Delta\mathbf{F} = \left\| \mathbf{F}_r(\mathbf{p}^{(i)}) - \mathbf{F}_r(\mathbf{p}^{(j)}) + \frac{\mathbf{F}_t(\mathbf{p}^{(i)})}{r(\mathbf{p}^{(i)})} - \frac{\mathbf{F}_t(\mathbf{p}^{(j)})}{r(\mathbf{p}^{(j)})} \right\| \tag{10}$$

If $\mathbf{p}^{(i)}$ and $\mathbf{p}^{(j)}$ are sufficiently close, $\mathbf{F}_r(\mathbf{p}^{(i)}) \approx \mathbf{F}_r(\mathbf{p}^{(j)})$ and $\mathbf{F}_t(\mathbf{p}^{(i)}) \approx \mathbf{F}_t(\mathbf{p}^{(j)})$. As a result the rotational component of flow cancels out in the spatial difference and:

$$\Delta\mathbf{F} \approx \left\| \mathbf{F}_t(\mathbf{p})\,\Delta\left(\frac{1}{r}\right) \right\| \tag{11}$$

That is, the difference in flow across the edge is proportional to the difference of the reciprocal of depth across the edge. The relationship between stereo disparity and depth is very similar to the relationship between optical flow and depth:

$$d(\mathbf{p}) = d_v(\mathbf{p}) + \frac{d_b(\mathbf{p})}{r(\mathbf{p})} \tag{12}$$

where $d(\mathbf{p})$ is the stereo disparity at $\mathbf{p}$, d_v is a term dependent on the camera vergence, and d_b is a term dependent on the baseline separating the cameras. Using the same argument as above, we have:

$$\Delta d \approx \left\| d_b(\mathbf{p})\, \Delta \left(\frac{1}{r} \right) \right\| \tag{13}$$

Over a local neighborhood, $\mathbf{F}_t$ and d_b will remain essentially constant, while $\Delta \frac{1}{r}$ will generally vary. Dividing equation (11) by equation (13) shows that the *ratio* of $\Delta\mathbf{F}$ to Δd remains constant, as long as the points over which the differences are taken are the same for flow and disparity.

Flow boundaries associated with moving objects are not subject to this constraint. As a result we can detect moving objects by looking for local neighborhoods over which the ratio $\Delta\mathbf{F}/\Delta d$ varies significantly. We never have to solve for the actual depth, nor do we need to know the functions $\mathbf{F}_t$, $\mathbf{F}_r$, d_v, or d_b. The solutiun does not depend on information about camera motion or relative camera geometry. For this approach to work, however, there has to be significant changes in depth over the background, not just between the background and any moving objects. There is reason to believe that such variation is important to a large class of moving object detection algorithms.

8 Examples.

All of the methods described in sections 6 and 7 have been tested experimentally. Four examples are presented below, all involving a moving camera and potentially moving objects. Two cases exploit the epipolar constraint. The first of these involves a situation in which camera rotation is known, but not camera translation. In the second case, a potentially moving object is being actively tracked. Results are also presented for two methods utilizing constraints resulting from the comparison of depth and flow. The simplest of these involves objects moving over a smooth environment. The final example compares flow and disparity across boundaries of possibly moving objects, using the technique of section 7.3.

Figure 4 shows the first frame in a sequence of of images of an outdoor scene. In this example, the camera rotates and translates with respect to the environment while the toy vehicle moves to the right between image frames. The rotational velocity of the camera with respect to the environment was measured. The optical flow field shown in figure 5 was obtained by the token matching technique described in [20]. The translational flow field shown in figure 6 was obtained by subtracting the rotational flow component computed from the known rotational velocity from the observed optical flow field (figure 5). The gradient of flow direction in the translational flow field was used to detect the boundaries of moving objects. Figure 7 shows the detected boundary of a moving object overlaid onto the first frame of figure 4.

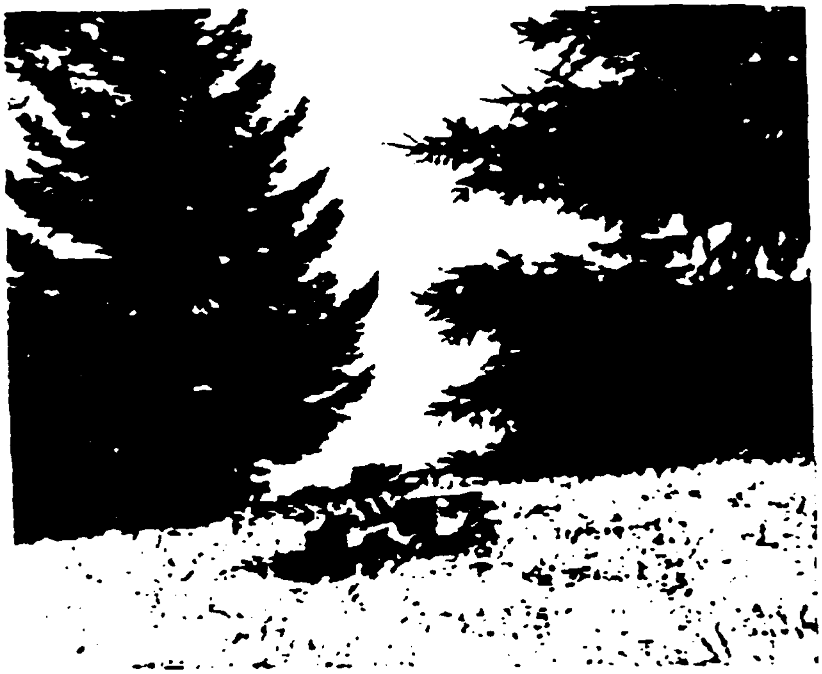


Figure 4: First frame of outdoor sequence.

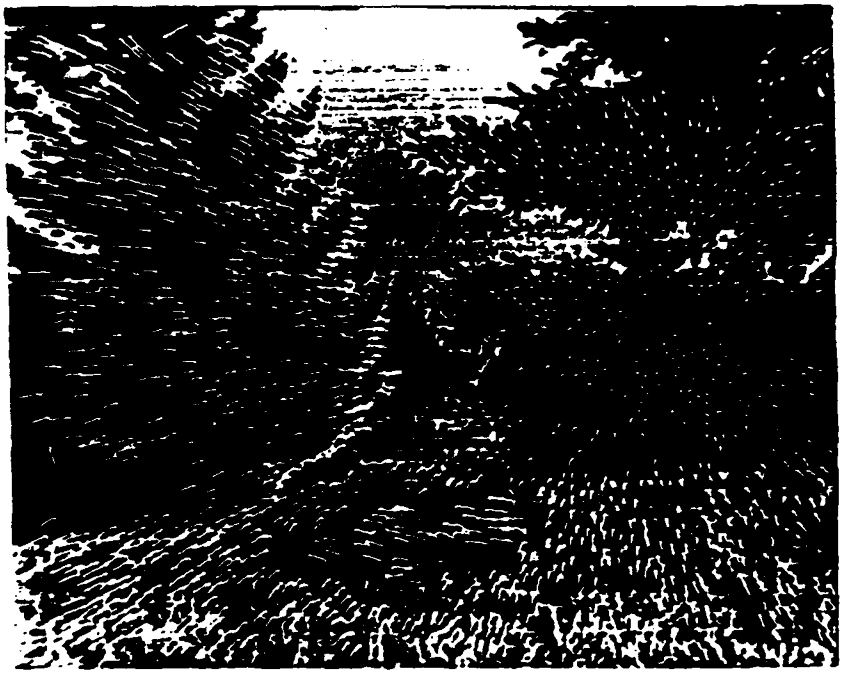


Figure 5: Optical flow field obtained from the image sequence of figure 4.

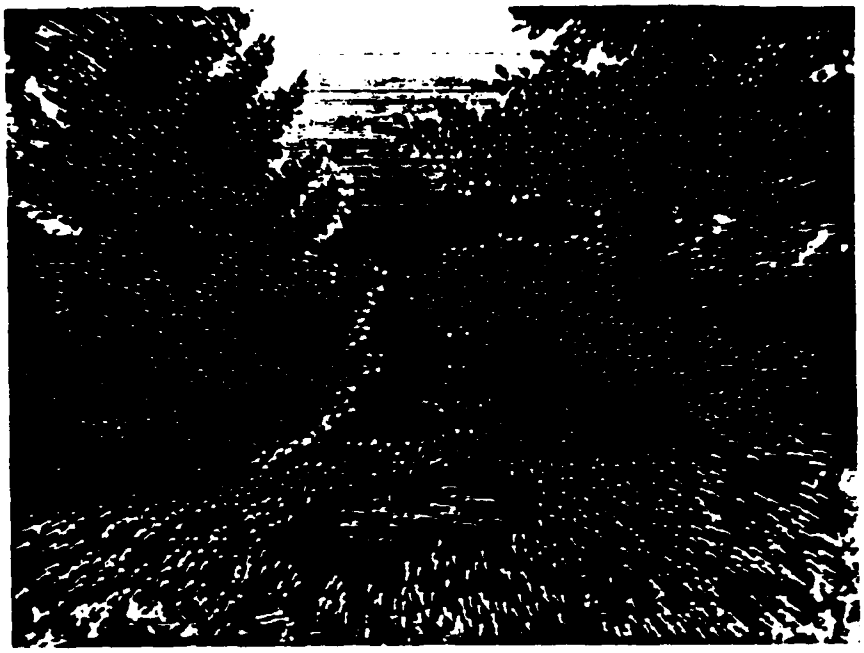


Figure 6: Translational flow field determined from the optical flow field of figure 5.

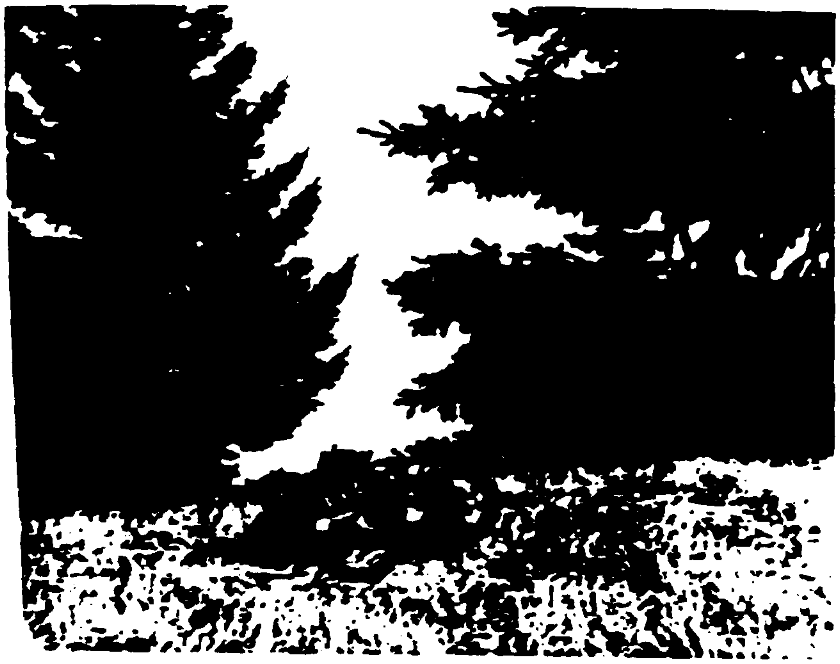


Figure 7: Boundary of a moving object overlaid onto the first image of figure 4.

In figure 8 the mechanical toy creature in the center of the image is being tracked by the camera while the camera is translating to the left with respect to the environment. Figure 9 shows the estimated optical flow. Figure 10 shows a histogram of the directions of the optical flow. Note that there are two distinct peaks in the histogram. The variation in flow direction over the image was computed to be approximately 34 degrees, indicating that the tracked object was in fact moving.

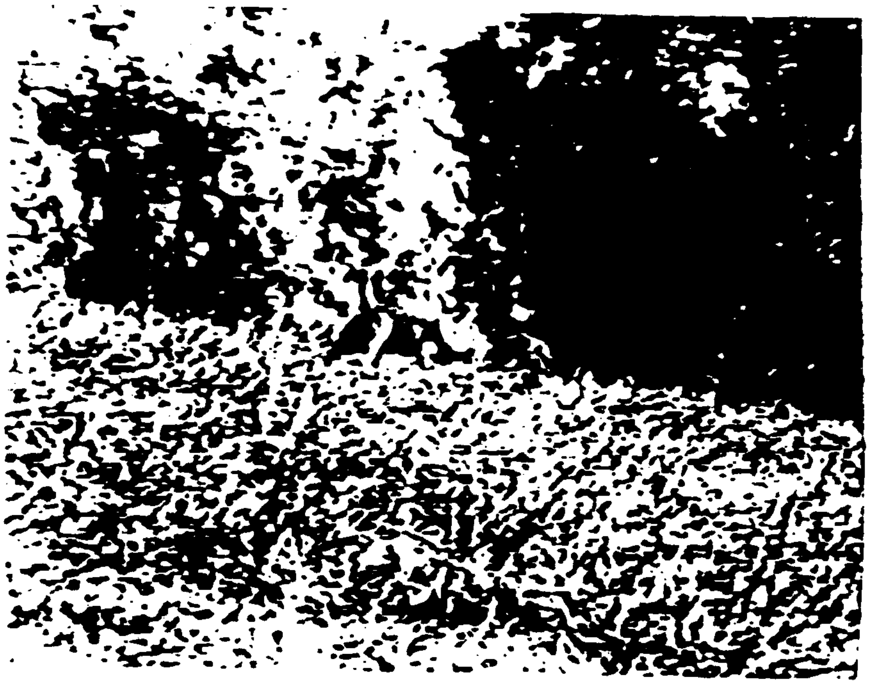


Figure 8: First frame of tracking sequence.

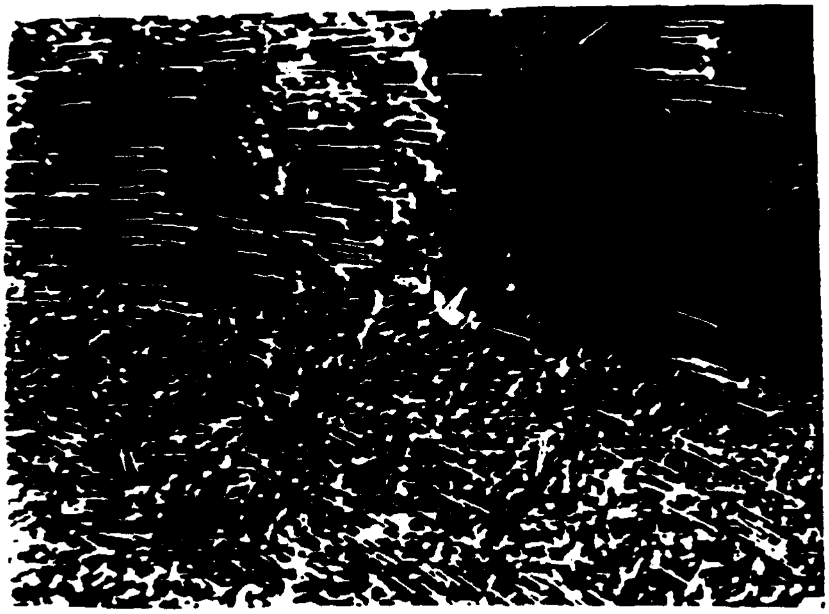


Figure 9: Optical flow field obtained from the image sequence of figure 8.

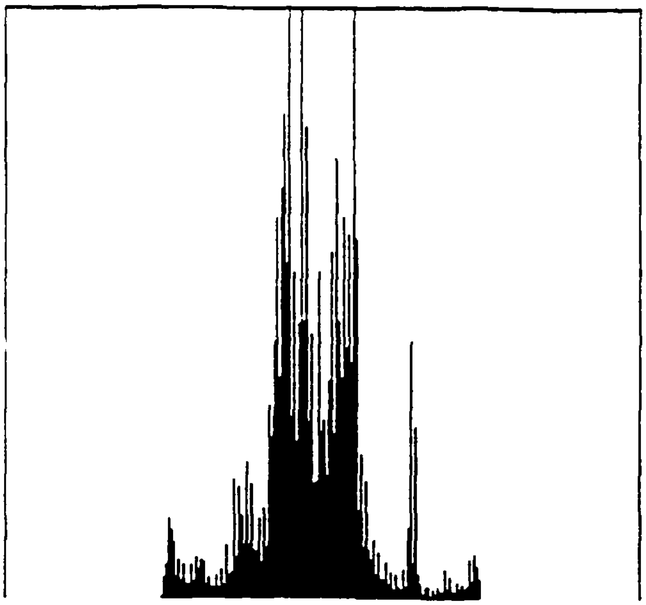


Figure 10: Histogram of the flow direction of the optical flow vectors in figure 9.

As a comparison, a similar experiment in which the tracked object, a rock, is stationary with respect to the environment while the camera is moving was also preformed. A pair of images similar to that of figure 8 were obtained. The resulting estimated optical flow field is shown in figure 11. Its corresponding histogram is shown in figure 12. Note that only one distinct peak is observed in this histogram. The global variation in flow direction in this case was computed to be approximately 11° which is significantly smaller than that of the previous example.

An image sequence starting with the frame shown in figure 13 is used to illustrate the technique for detecting objects moving in a smooth environment. In this example, the camera moves with respect to an environment consisting of various small pieces of hardware lying on a planar surface. The optical flow field shown in figure 14 was obtained in the same manner as in figure 5. Figure 15 shows the locations of large variations in optical flow values, corresponding to the boundary of a moving object.

A stereo image sequence starting with the stereo pair shown in figure 16 is used to illustrate the technique of indirect comparison of flow and disparity as a basis for moving object detection. Both the flow field shown in figure 17, and the disparity field shown in figure 18 were obtained using the method of figure 5. Comparing the ratio of the change in disparity values to the change in flow values across neighboring points, and selecting as the boundaries of moving objects those areas in which there is a distinct discontinuity in that ratio, results in the identification of the boundaries indicated in figure 19.

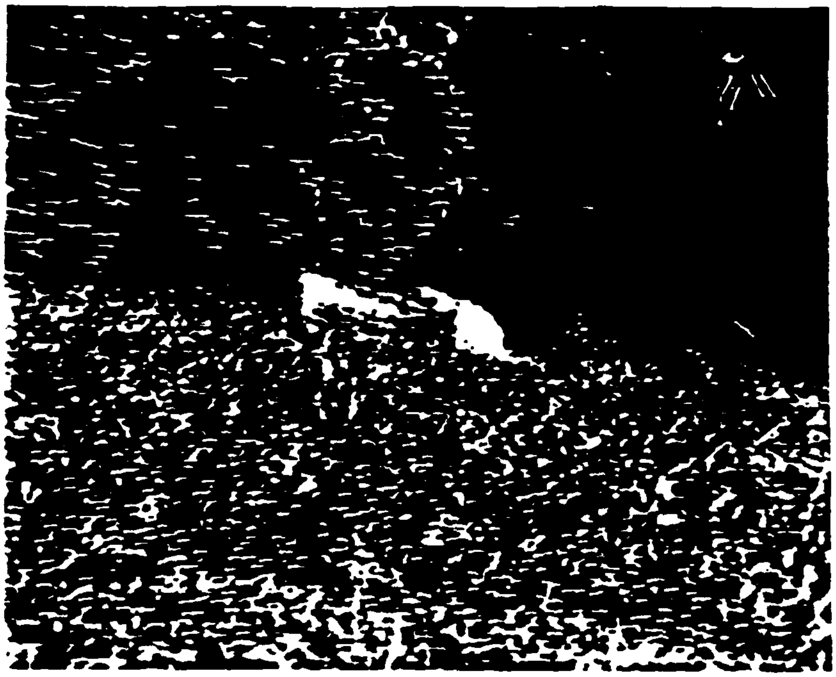


Figure 11: Optical flow field obtained from tracking an object which is stationary with respect to the environment.

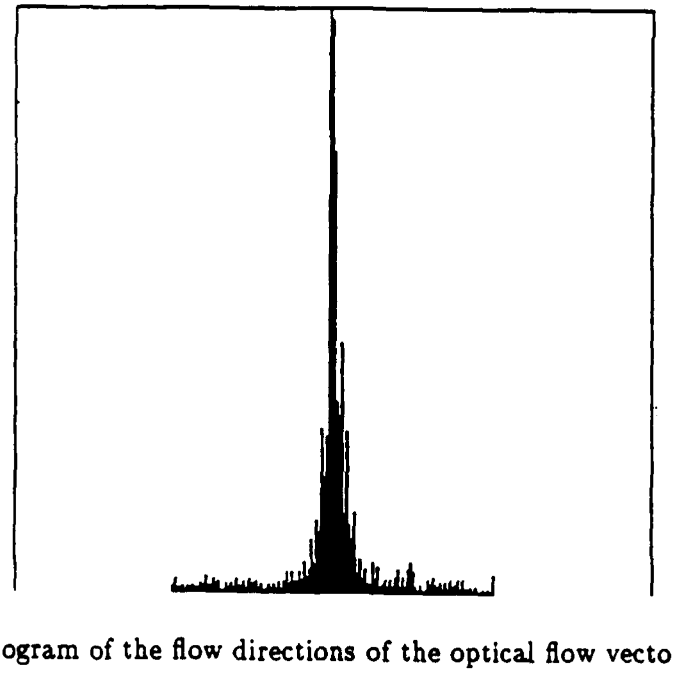


Figure 12: Histogram of the flow directions of the optical flow vectors in figure 11.

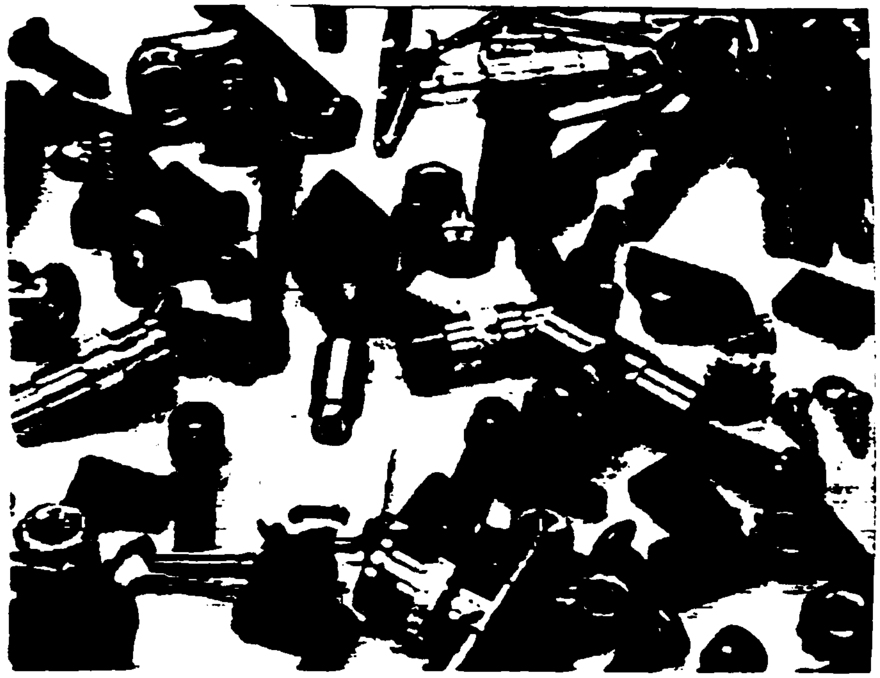


Figure 13: First frame, miscellaneous hardware sequence.

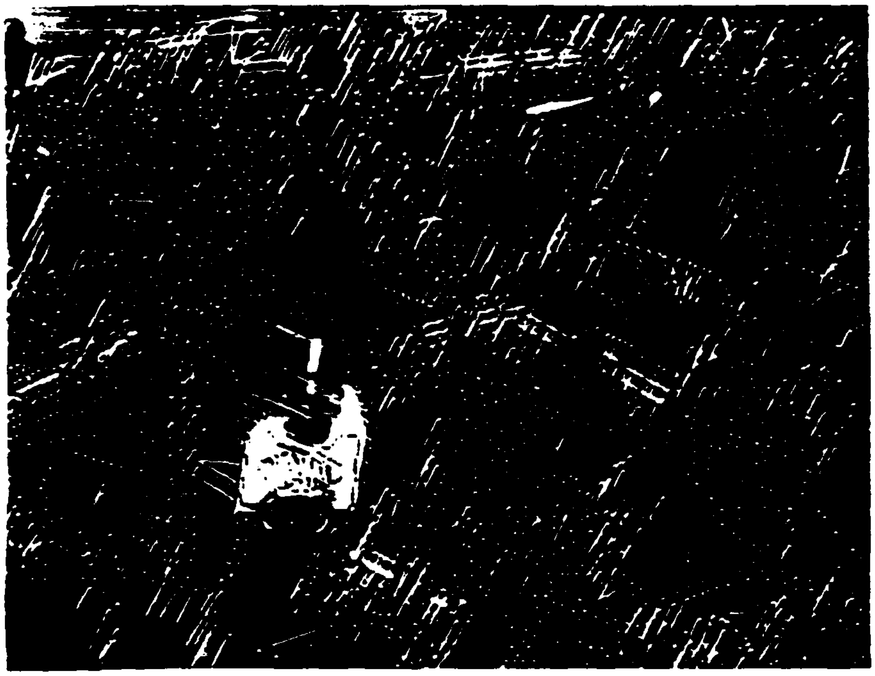


Figure 14: Optical flow field obtained from the image sequence of figure 13.

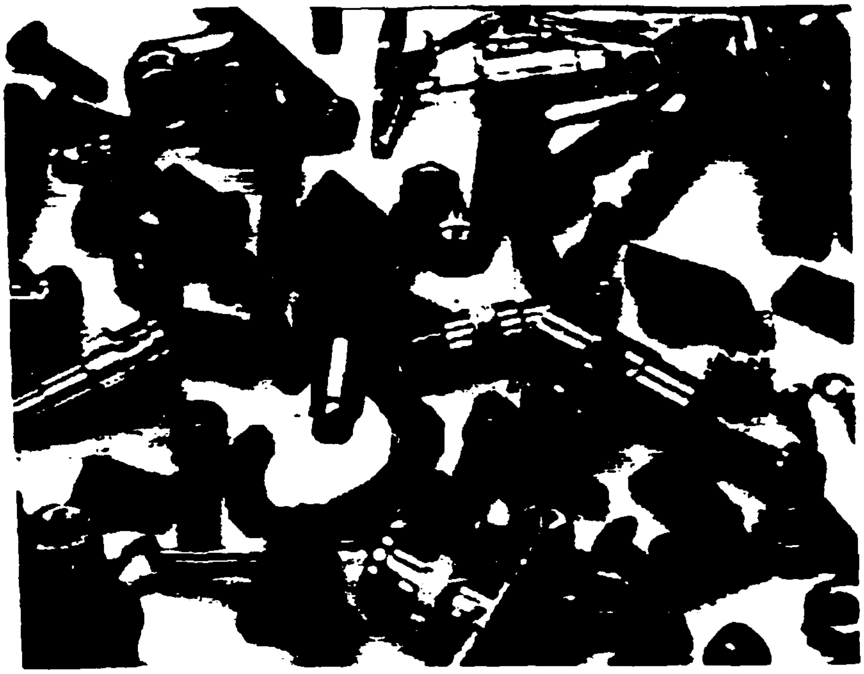


Figure 15: Boundary of a moving object overlaid onto the first image of figure 13.

Figure 16: First pair of stereo images in a sequence.

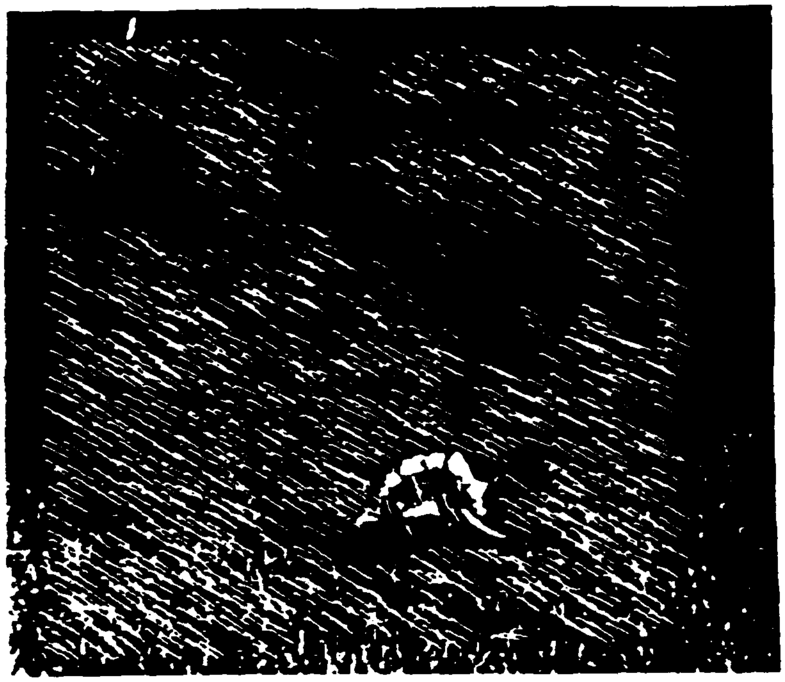


Figure 17: Optical flow field obtained for right image sequence of figure 16.

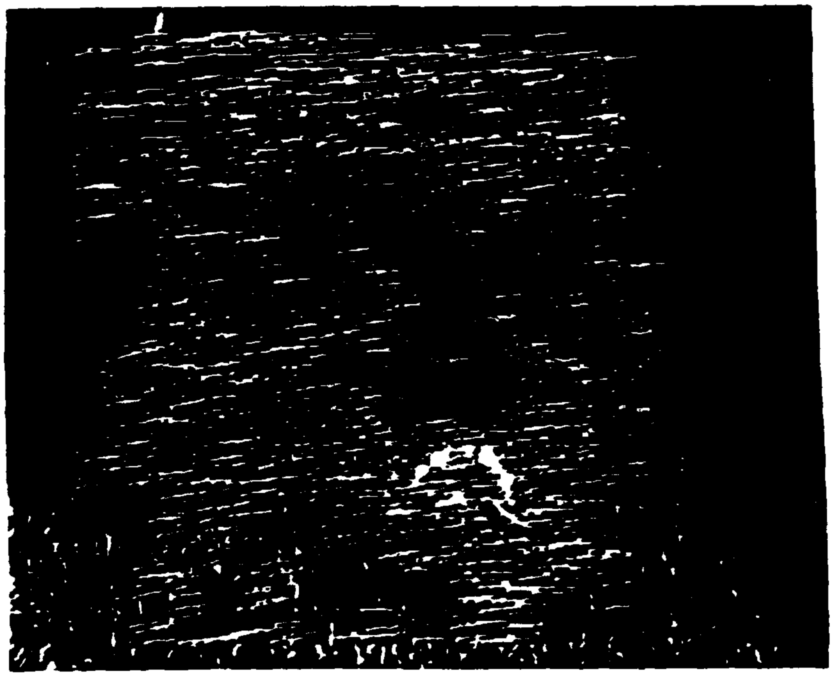


Figure 18: Disparity field obtained across the stereo pair in figure 16.

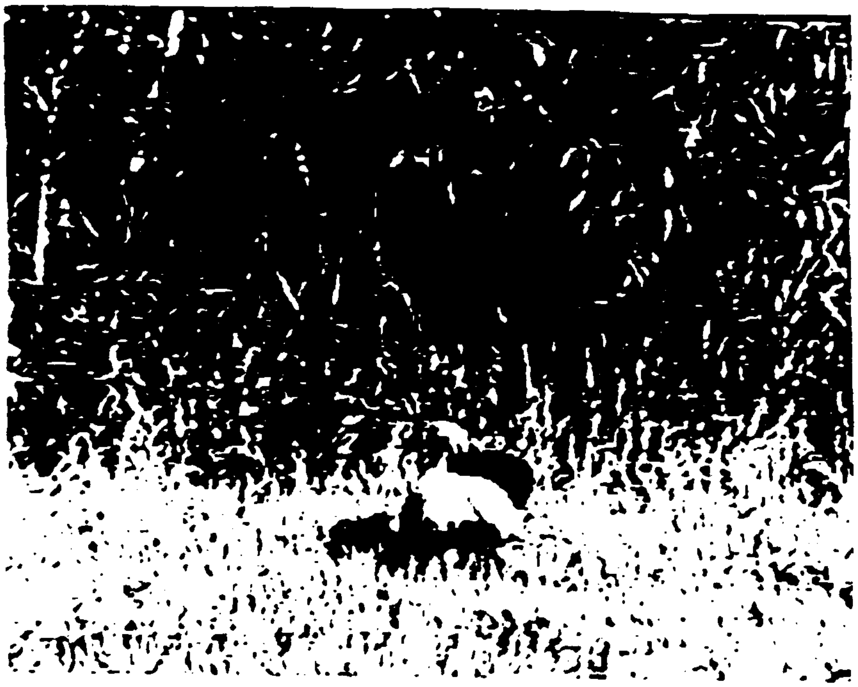


Figure 19: Boundary of a moving object overlaid onto the right image of the stereo pair in figure 16.

9 Discussion.

9.1 Which method to use?

This paper presents a collection of loosely related techniques for visually detecting moving objects. Detection based purely on visual motion from a single camera seems quite difficult. Each of the methods presented here uses some sort of additional information, either about current camera motion or scene structure. The methods are characterized by the additional information used, the underlying constraints exploited, and the particular computational structure used to implement the technique. It is likely that reliable moving object detection will require several complimentary techniques, along with a method for selecting which detector to trust in any particular situation.

9.2 Computational structure.

The methods described above can be grouped into three classes. *Point-based* techniques (completely known motion) compare individual optical flow vectors against some standard to determine incompatibilities with the motion of the camera relative to the environment. In all cases described here, the compatibility measure is based on a directional constraint associated with the focus of expansion of the translational flow field. Point-based methods have the advantages of computational simplicity and the ability to detect very small moving objects. They will be most effective when parameters of motion are known precisely and the magnitude of the translational flow field at the point in question is sufficiently large to allow an accurate estimate of direction. *Edge-based* techniques (known rotation, smooth surface) roughly correspond to traditional edge detection.

Edge-based motion detection is characterized by the differential flow properties examined and by the filtering technique used to separate edges due to range discontinuities from those due to moving objects. The approach is effective when surfaces are smooth and techniques exist for accurately locating those range discontinuities that do exist. Edge-based methods have the advantage of specifying the outline of moving objects that are detected. They are likely to be of limited use when moving objects are quite small. *Region-based* techniques (tracked object, depth/flow comparisons) examine optical flow values over a region, searching for distributions incompatible with rigid motion. As with edge-based approaches, the viewed region must include portions of both object and environment. As long as the region includes portions of both object and environment, this is an effective test for moving objects that does not require any information about camera motion. The region-based method based on tracking potentially moving objects does not require any information about camera motion, but does require that there be significant variations in range over the visible portions of the environment.

9.3 Limitations.

All detection algorithms founded on the motion epipolar constraint share two important shortcomings. First, environmental flow vectors will be small near the FOE regardless of the ranges involved. As a result, detection based on flow orientation will be unreliable within a region around the FOE.[4] This means that epipolar-based methods will have difficulties for viewing directions close to the direction of motion. This is of course the direction in which moving object detectior is likely to be most important. One heuristic for partially overcomming limitations near the FOE is to look for large magnitude values of translational flow near the FOE. Such values correspond either to moving objects or to environmental points that are very close to the camera. Secondly, while the motion epipolar methods were developed to allow for the *possibility* of a moving camera, translational camera motion is actually a *requirement*. Without translational motion, there is no motion epipolar constraint to violate. More specifically, not only must the camera be moving, but significant portions of the visible environment must be sufficiently close to generate detectable non-zero translational flow values. Most methods based on the depth/flow or rigidity constraints should work for both moving and stationary cameras.

No method for detecting moving objects will be effective if it depends on knowing precise values of optical flow. Techniques for estimating optical flow are intrinsically noisy (e.g., see [22]). Additional difficulties arise due to the idealized nature of equations (1–3). Real cameras are not point projection systems. Substantial effort is required to accurately determine the values of x and y in (1–3). Geometric distortions in the optical and sensing systems affect measured locations on the image plane. Variabilities in effective focal length can be substantial. Reliable techniques will be based on searching for large magnitude effects in the flow field [23]. All of the methods described above compare flow vectors to some predetermined standard, or look for significant differences across flow boundaries. As a result, all deal with relatively large magnitude effects. Reliability is

[4] Lawton talks about a "dead zone" around the FOE within which no information based exclusively on camera motion is available [21]. This effect is a problem not only for moving object detection, but also for techniques such as motion stereo.

still dependent on scene structure, the nature of camera motion, and position in the visual field relative to the direction of translation.

Acknowledgement.

Martin Kenner provided significant assistance in preparing the examples.

References

[1] G. Adiv, “Inherent ambiguities in recovering 3-d motion and structure from a noisy flow field”, *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 70–77, 1985.

[2] *Proc. Workshop on Motion: Representation and Analysis*, May 1986.

[3] S. Ullman, *The Interpretation of Visual Motion*, MIT Press, Cambridge, MA, 1979.

[4] R. Jain, W.N. Martin, and J.K. Aggarwal, “Extraction of moving object images through change detection”, *Proc. Sixth International Joint Conference on Artificial Intelligence*, 425–428, 1979.

[5] R. Jain, D. Militzer, and H.-H. Nagel, “Separating non-stationary from stationary scene components in a sequence of real world TV images”, *Proc. Fifth International Joint Conference on Artificial Intelligence*, 425–428, 1977.

[6] A.M. Waxman and J.H. Duncan, “Binocular image flows”, *Proc. Workshop on Motion: Representation and Analysis*, 1986.

[7] W.B. Thompson, K.M. Mutch, and V.A. Berzins, “Dynamic occlusion analysis in optical flow fields”, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-7:374–383, July 1985.

[8] R.C. Jain, “Segmentation of frame sequences obtained by a moving observer”, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-6:624–629, September 1984.

[9] W.B. Thompson, “Combining motion and contrast for segmentation”, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-2:543–549, November 1980.

[10] W.F. Clocksin, “Perception of surface slant and edge labels from optical flow: a computational approach”, *Perception*, 9:253–269, 1980.

[11] K. Nakayama and J.M. Loomis, “Optical velocity patterns, velocity sensitive neurons, and space perception: a hypothesis”, *Perception*, 3:63–80, 1974.

[12] A.R. Bruss and B.K.P. Horn, “Passive navigation”, *Computer Vision, Graphics and Image Processing*, 21(1):3–20, January 1983.

[13] D.H. Ballard and O.A. Kimball. "Rigid body motion from depth and optical flow". *Computer Vision. Graphics and Image Processing*, 22:95–115. 1983.

[14] D.A. Marr. *Vision*. W.H. Freeman. San Francisco. 1982.

[15] A. Bandopadhay. B. Chandra. and D.H. Ballard. "Active navigation: tracking an environmental point considered beneficial", *Proc. Workshop on Motion: Representation and Analysis*. 23–29. 1986.

[16] T.S. Huang. S.D. Blostein. A. Werkheiser. M. McDonnel. and M. Lew. "Motion detection and estimation from stereo image sequences: some preliminary experimental results". *Proc. Workshop on Motion: Representation and Analysis*. 45–46. 1986.

[17] J.J. Gibson. *The Perception of the Visual World*, Riverside Press, Cambridge MA. 1950.

[18] R.A. Brooks. A.M. Flynn, and T. Marill, "Self calibration of motion and stereo for mobile robots", *Proc. 4th Int. Symposium on Robotics Research*, 1987.

[19] J.H. Reiger and D.T. Lawton. "Sensor motion and relative depth from difference fields of optic flows", *Proc. Eighth International Joint Conference on Artificial Intelligence*, 1027–1031. 1983.

[20] S.T. Barnard and W.B. Thompson. "Disparity analysis of images", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-2:333–340, July 1980.

[21] D.T. Lawton. personal communication.

[22] J.K. Kearney, W.B. Thompson. and D.L. Boley, "Optical flow estimation: an error analysis of gradient-based methods with local optimization", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-9:229–244, March 1987.

[23] W.B. Thompson and J.K. Kearney, "Inexact vision", *Proc. Workshop on Motion: Representation and Analysis*, 15–21, 1986.

1st INT. CONFERENCE ON COMPUTER VISION (ICCV) '87

DETECTING MOVING OBJECTS

William B. Thompson *Ting-Chuen Pong*

Computer Science Department
University of Minnesota
Minneapolis, MN 55455

ABSTRACT

The detection of moving objects is important in many tasks. This paper examines moving object detection based primarily on visual motion. We conclude that in realistic situations, detection using visual information alone is quite difficult, particularly when the camera is also moving. The availability of additional information about camera motion and/or scene structure greatly simplifies the problem. We develop detection algorithms for the cases in which 1) camera motion is known, 2) only camera rotation is known, 3) only camera translation is known, 4) objects move in contact with a smooth surface, and 5) an object is being actively tracked, but the camera motion associated with the tracking is not known precisely. Examples of several of these techniques are presented.

1. Introduction.

One important function of a vision system is to recognize the presence of moving objects in a scene. If the camera is stationary and illumination constant, this can be done by simple techniques which compare successive image frames, looking for significant differences. If the camera is moving, the problem is considerably more complex. For the purposes of this discussion, *moving objects* are taken to be any objects moving with respect to the stationary portions of the scene, which we refer to as the *environment*. For a moving camera, both moving objects and stationary portions of the scene may be changing position with respect to the camera and thus generating visual motion in the imagery. A moving camera leads to difficulties because of the need to determine objects moving with respect to the environment, rather than the much easier problem of finding objects moving with respect to the camera. In this paper, we deal with the problem of detecting moving objects from a moving camera based on optical flow.

The visual detection of moving objects is a surprisingly difficult task. A simple example illustrates just how serious the problem can be. Consider the optical flow field shown in figure 1, which appears to show a small, square region in the center of the image moving to the right and surrounded by an apparently stationary background. Such a flow field can arise from several equally plausible situations: 1) The camera is stationary with respect to the environment, and the central region corresponds to an object moving to the right. 2) The camera is moving to the left with respect to the environment, most of the environment is sufficiently distant so that the generated optical flow is effectively zero, while the central region corresponds to a surface near to the camera but stationary with respect to the environment. 3) The camera and object are moving with respect to both the environment and each other, though the environment is sufficiently distant so that there is no perceived optical flow. It is not possible to tell whether or not this seemingly simple pattern corresponds to a moving object!

Figure 1 provides one example of why a general and reliable solution to the problem of moving object detection based only on visual motion is not feasible. Robust solutions require that additional information about camera motion and/or scene structure be available. In this paper, we examine a variety of types of information that might be available. Each information source places constraints on the optical flow fields that can be generated by a camera moving through an otherwise static environment. Violations of these constraints are thus necessarily due to moving objects.

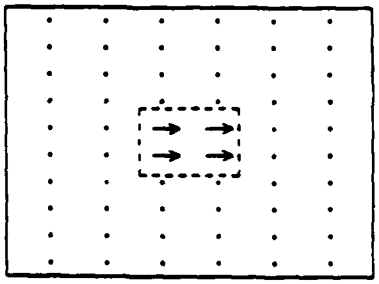


Figure 1: Is The Central Region a Moving Object?

Figure 2 summarizes potential sources of information and the associated constraints on optical flow. The next section lists general properties needed by reliable detection algorithms. Following this is a derivation of each of the flow constraints. We conclude with experimental demonstration of several of the techniques and general observations about the nature of these methods.

2. Assumptions.

We start with the presumption that motion detection algorithms should be designed with the following properties in mind:

The field of view may be relatively narrow.

Motion detection should not depend on the use of wide angle imaging systems. Such systems may not be available in a particular situation, and if used may increase the difficulty or recognizing small moving objects. As a result, detection algorithms should not depend on subtle properties of perspective.

This work was supported by Air Force Office of Scientific Research contract AFOSR-85-0382.

CH2465-3/87/0000/0201$01.00 © 1987 IEEE

Knowing:	Yields a constraint on:
full parameters of motion	flow values
parameters of rotation	variability of flow direction
surfaces are smooth	local variability of direction or magnitude of flow
object is being tracked	global variability of direction of flow

Figure 2: Constraints on Flow.

The image of moving objects may be small with respect to the field of view.

This is clearly desirable for reliability. Moving objects may be far away and subtended by relatively small visual angles. We need methods capable of identifying single image points, or at least small collections of points, as corresponding to moving objects. Detection algorithms thus cannot depend on variations in flow over a potentially moving object.

Only monocular imagery is available.

This is equivalent to the situation where objects of interest can be far away relative to the camera base-line in a stereo viewing situation.

Estimated optical flow fields will be noisy.

No method is capable of estimating optical flow with arbitrary accuracy. Motion detection based on optical flow must be tolerant of noisy input.

Only "instantaneous" optical flow is used.

A restriction to instantaneous flow eliminates the use of temporal derivatives of flow and/or multiple views at distinct time intervals. Temporal differentiation will increase noise in the estimated flow values. Use of multiple views increase computational complexity. (In fact, experience with There are reasons to believe that multi-frame analysis techniques may in fact improve reliability [1], though they are not examined in this work.)

3. Constraints on Optical Flow.

The basic mathematics governing the optical flow generated by a moving camera is well known. We take our notation from [2], using a coordinate system fixed to the camera (e.g. the world can be thought of as moving by a stationary camera). Optical flow values are a function of image location, the relative motion between the camera and the surface point corresponding to the image location, and the distance from the camera to the corresponding surface point. Let $\mathbf{p} = (x, y)$ refer to an image location, where x and y have been normalized by the focal length of the camera. Let $\mathbf{P} = (X, Y, Z)$ be the coordinates of the surface point projecting onto (x, y), specified in a coordinate system with origin at the camera and Z axis along the optical axis of the camera. Specify the motion of the point at (X, Y, Z) with respect to the camera in terms of a translational velocity $\mathbf{T} = (U, V, W)^T$ and a rotational velocity $\boldsymbol{\omega} = (A, B, C)^T$. The optical flow, $\mathbf{B} = (u, v)$, at $\mathbf{p}$ is purely a function of x, y, $\mathbf{T}$, $\boldsymbol{\omega}$, and Z:

$$u = u_t + u_r \quad , \quad v = v_t + v_r \tag{1}$$

where u is the x component of flow, v is the y component of flow, and

$$u_t = \frac{-U + xW}{Z} \ , \quad v_t = \frac{-V + yW}{Z} \tag{2}$$

$$\begin{aligned} u_r &= Axy - B(x^2 + 1) + Cy \\ v_r &= A(y^2 + 1) - Bxy - Cx \end{aligned} \tag{3}$$

Let the parameters specifying camera motion with respect to the environment be $\mathbf{T}_e$ and $\boldsymbol{\omega}_e$ and the corresponding parameters specifying relative motion between the camera and a scene point $\mathbf{P}$ be $\mathbf{T}_P$ and $\boldsymbol{\omega}_P$.

3.1. Known translation and rotation.

The parameters of camera motion constrain possible optical flow values that can occur due to camera motion with respect to the environment.

If complete information about instantaneous camera motion is available, then $\mathbf{T}_e$ and $\boldsymbol{\omega}_e$ are known. If the camera is translating but not rotating with respect to the background, $\boldsymbol{\omega}_e = 0$ and all flow vectors due to the moving image of the background will radiate away from a *focus of expansion* (FOE). From equation (1), it is easy to see that the image plane location of the FOE is at:

$$x_{foe} = \frac{U}{W} \ , \quad y_{foe} = \frac{V}{W} \tag{4}$$

The location of the FOE depends only on the direction of translation, not on the speed, so methods for motion detection which depend on the location of the FOE do not actually require the complete parameters of translational motion. The FOE may not lie within the visible portion of the image (and in fact may be a focus of contraction). A FOE at ∞ corresponds to pure lateral motion, which generates a parallel optical flow pattern. At every image point $\mathbf{p}$, knowing the FOE fully specifies the direction of optical flow associated with any surface point stationary with respect to the environment. At $\mathbf{p}$:

$$\theta_{foe} = \tan^{-1} \frac{V - Wy}{U - Wx} \ , \quad \theta_{flow} = \tan^{-1} \frac{v_t}{u_t} \tag{5}$$

where θ_{foe} is the direction from $\mathbf{p}$ towards the FOE and θ_{flow} is the direction of optical flow at $\mathbf{p}$. (Note that the first equation is still well defined even if $W = 0$, corresponding to a focus of expansion at ∞ in image coordinates.) Any flow values with a different direction correspond to moving objects [3]. E.g., moving objects exist whenever $|\theta_{foe} - \theta_{flow}| > \varepsilon$, for some appropriate ε. (It is possible that moving objects coincidentally generate flow values compatible with this constraint.) This approach requires the estimation of only the direction of flow, not either the magnitude or spatial variation of flow.

Camera rotation introduces considerable complexity. Knowledge of camera motion no longer constrains the direction of background flow. Nevertheless, at a given point $\mathbf{p}$, flow is constrained to a one-dimensional family of possible vector values. The family is given by (1) – (3) where Z ranges over all positive values. The analysis can be simplified because of the linear nature of (1). u_r and v_r depend only on the parameters of rotation and not on any shape property of the environment. Because the values of u_r and v_r at a particular point p do not depend on Z, they can be predicted knowing only ω. These values can be subtracted from the observed optical flow field, leaving a *translational flow field*:

$$\mathbf{F}_t = (u_t\ ,v_t) = \mathbf{F} - \mathbf{F}_r\ ,\quad \mathbf{F}_r = (u_r\ ,v_r) \tag{6}$$

where u_r and v_r are defined in equation (3). This field behaves just as if no rotation was occurring, and thus moving objects can be located using the FOE technique described above. For the remainder of this paper, when rotation is present, we will take the term FOE to refer to the focus of expansion of this translational field.

In principle, even if camera motion is not known $\mathbf{T}_e$ and ω_e may be estimated from the imagery [2], subject to a positive, multiplicative scale factor for $\mathbf{T}_e$. Two serious problems exist, however. Narrow angles of view make estimation of camera motion difficult, as significantly different parameters of motion and surface shape can yield nearly identical optical flow patters [4]. In addition, techniques such as [2] uses a global minimization approach which will not perform well if moving objects make up a substantial portion of the field of view. A clustering approach (e.g. [5]) can be made tolerant of the moving objects, though great difficulty can be expected dealing with a five dimensional cluster space.

3.2. Known rotation.

The parameters of camera rotation constrain the local variability of optical flow direction that can occur due to camera motion with respect to the environment.

Often, information about camera rotation is available, even when the direction of translation is not known. Non-visual information about camera motion often comes from inertial sources. Such sources are much more accurate in determining rotation than translation. Rotation involves a continuous acceleration which is easily measured. The determination of translation requires the integration of accelerations, along with a starting boundary value. Errors in estimated translation values rapidly accumulate. A simple technique allows the detection of moving objects when only camera rotation is known.

In the previous sections, knowledge of camera rotation made it possible to compute the translational flow field, $\mathbf{F}_t$. Knowledge of translation was then used to locate the FOE and thus constrain the direction of flow vectors associated with the environment. If only rotation is known, then it is still possible to determine the translational flow field, but not the FOE. Visual methods could be applied to the translational flow field to estimate the location of the FOE, but these methods suffer from a number of practical limitations when applied to noisy data. An alternate approach can be used which does not require the prior determination of the FOE. The translational flow field extends radially from the focus of expansion. At any point significantly away from the FOE, the direction of flow (but not necessarily the magnitude of flow) will vary slowly. Directional variability can be evaluated based on equation (5):

$$\begin{aligned} \frac{\delta\,\theta_{foe}}{\delta x} &= \frac{W(V-yW)}{(V-yW)^2+(U-xW)^2} \\ \frac{\delta\,\theta_{foe}}{\delta y} &= -\frac{W(U-xW)}{(V-yW)^2+(U-xW)^2} \end{aligned} \tag{7}$$

The gradient of the direction of the translational flow field can thus be obtained as

$$\left(\frac{\delta\theta_{foe}}{\delta x}\right)^2 + \left(\frac{\delta\theta_{foe}}{\delta y}\right)^2 = \frac{1}{(y_{foe}-y)^2+(x_{foe}-x)^2} \tag{8}$$

where $(x_{foe}\ y_{foe})$ is the image plane location of the FOE. We can see from the above equation that over any local area away from the FOE, variations in the *direction* of the translational flow field will be small. Flow arising due to moving objects is of course not subject to this restriction. The gradient of flow field direction can thus be used to detect the boundaries of moving objects. At these boundaries, flow direction will vary discontinuously[1].

A complementary technique is available to deal with situations in which translation but not rotation is known. We can expect these situations to be rare, however. If the direction of translation were known over some interval of time, it would be an easy matter to determine the rotation by examining the rate of change of direction.

3.3. Motion over smooth surfaces.

Object motion over smooth surfaces constrains the local variability of flow.

Knowledge of the shape of environmental surfaces can be used to simplify the motion detection problem. Scene structure may be known precisely (e.g. the range to visible surface points) or in terms of general properities (e.g. significant depth discontinuities can be expected). Information about scene structure can come from visual sources (e.g stereo [9,10]), or from pre-existing models of the environment. If both the optical flow, $(u\ ,v)$, and the depth, Z, are known for a collection of surface points in the environment, then (1) – (3) can be used to create a system of equations which can be solved for the parameters of motion $\mathbf{T}$ and ω. If the collection of points includes some values associated with the environment and others associated with one or more objects moving with respect to the environment, the system of equations used to solve for $\mathbf{T}$ and ω will be inconsistent. Checking the system for consistency can therefore be used as a test for the presence of a moving object (e.g. a test for non-rigid motion in the field of view.)

If moving objects must remain in contact with environmental surfaces (e.g. vehicular motion), a less complex technique depending only on knowing the image plane locations corresponding to discontinuities in range is possible. If no objects are moving within the field of view, equations (1) – (3) can be simplified into the following form:

$$flow(\mathbf{p}) = f_r(\mathbf{p}) + \frac{f_t(\mathbf{p})}{r(\mathbf{p})} \tag{9}$$

where at an image point p, *flow(p) is the optical flow* (a two-dimensional vector), f_r is the component of the flow due to the rotation of the scene with respect to the sensor, f_t is dependent on the translational motion of the sensor and the viewing angle relative to the direction of translation, and r is the distance between the sensor and the surface visible at p (i.e. the value of Z in equation 2 corresponding to the image location p). For fixed p, flow varies inversely with distance. Both f_r and f_t vary slowly (and continuously) with p. *Discontinuities in flow* thus correspond to discontinuities in r. This relationship holds only for relative motion between the camera and a single, rigid structure. When multiple moving objects are present, equation (9) must be modified so that

[1] Marr [6] claims "if direction of [visual] motion is ever discontinuous at more than one point – along a line, for example, – then an object boundary is present." Note that this is only necessarily true if no camera rotation is occurring (or equivalently, if camera rotation has been normalized by using the translational flow field).

there is a separate $f_r^{(i)}$ and $f_t^{(i)}$ specifying the relative motion between the sensor and each rigid object. Discontinuities in flow can now arise either due to a discontinuity in range or due to the boundaries of a moving object. If independent information is available on the location of range discontinuities, and other discontinuities in flow must be due to moving objects.

The motion detection problem becomes particularly simple if the environment is planar. In this case, depth discontinuities are not possible and *any* discontinuity in flow (either direction or magnitude) corresponds to the boundary of a moving object. Note that it is not sufficient to know simply that the environment is a "smooth" surface. From some viewing positions, even smooth surfaces may exhibit range discontinuities.

3.4. Tracking regions of interest.

Tracking an object constrains the global variability of the direction of flow in the surrounding area.

A vision system which can actively control camera direction is capable of tracking regions of interest over time, keeping some particular object centered within the field of view. Tracking regions of interest is desirable for many reasons other than the detection of moving objects (e.g. [11]), though the analysis of imagery arising from a tracking camera has not received much study by the computer vision community. If there are significant variations in depth over the visible portion of the background and if moving objects are relatively small with respect to the field of view, then moving object detection based on tracking can be accomplished without any actual knowledge of camera motion. (For motion detection, the tracking can easily be simulated if the camera is not actively controllable.)

If an object is being tracked, then its optical flow is zero. Flow based methods for determining whether or not a tracked object is moving must depend wholly on the patterns of flow in the background. Object tracking helps in moving object detection because it minimizes many of the difficulties due to rotation. When dealing with instantaneous flow fields, we can decompose the problem by considering all translational motion to be due to movement of the camera platform and all rotational motion due to pan and tilt of the camera to accomplish the tracking. (We will disregard any effects due to spin around the line of sight.) Consider the effect of tracking a point that is in fact part of the environment. The translational component of motion induces an optical flow pattern field extends radially from the focus of expansion, with magnitudes dependent on the range to the corresponding surface points. Over a local area away from the focus of expansion, the *direction* of translational flow will be approximately constant. The rotational component of motion induces a flow pattern which over a local area is approximately constant in both direction and magnitude. The magnitude and direction are exactly opposite the translational flow of the tracked point. From equations (2) and (3), it is easy to see that at the tracked point $(x,y) = (0,0)$

$$u_t = -\frac{U}{Z}, \quad v_t = -\frac{V}{Z} \tag{10}$$

$$u_r = -B, \quad v_r = A \tag{11}$$

Since the optical flow is zero at the tracked point, we have

$$-\frac{U}{Z} - B = 0, \quad or \quad u_t = -u_r \tag{12}$$

$$-\frac{V}{Z} + A = 0, \quad or \quad v_t = -v_r \tag{13}$$

The effect on the combined fields is that in the neighborhood of the tracked point, the direction of flow will be approximately constant (modulo 180°), with a magnitude dependent on the difference between the range to the corresponding surface point and the range to the tracked point. Now, consider tracking a point that is moving with respect to the environment. If environmental surface points are visible in the neighborhood of the tracked point, and if there is a variation in range to these environmental points, then there will be a variation in direction of flow over the neighborhood.

4. Examples.

A set of experiments on moving object detection based on the techniques discussed in the previous sections have been preformed on real images. Experimental results are presented in this section for the cases in which 1) the camera rotation is known, 2) objects move in a smooth environment, and 3) a potentially moving object is being actively tracked.

Figure 3 shows the first frame in a sequence of of images of an indoor scene. In this example, the camera rotates and translates with respect to the environment while the toy vehicle on the table moves to the right between image frames. The rotational velocity of the camera with respect to the environment was measured. The optical flow field shown in figure 4 was obtained by the token matching technique described in [10]. The translational flow field shown in figure 5 was obtained by subtracting the rotational flow component computed from the known rotational velocity from the observed optical flow field (figure 4). The gradient of flow direction in the translational flow field was used to detect the boundaries of moving objects. Figure 6 shows the detected boundary of a moving object overlaid onto the first frame of figure 3.

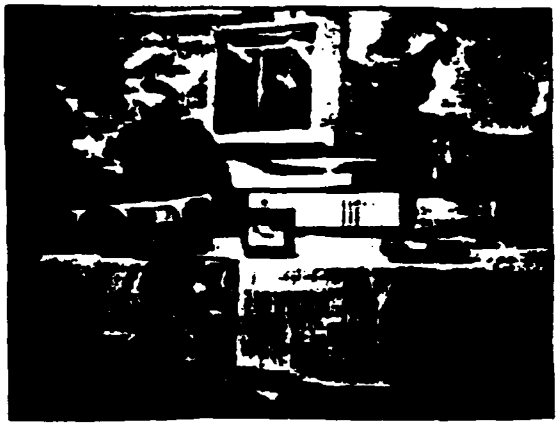


Figure 3: First frame of indoor scene.

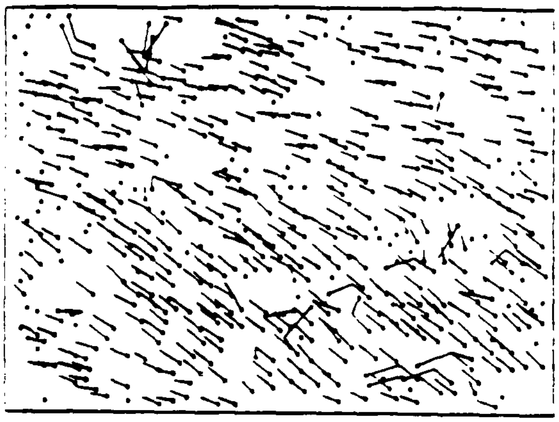


Figure 4: Optical flow field obtained from the image sequence of figure 3.

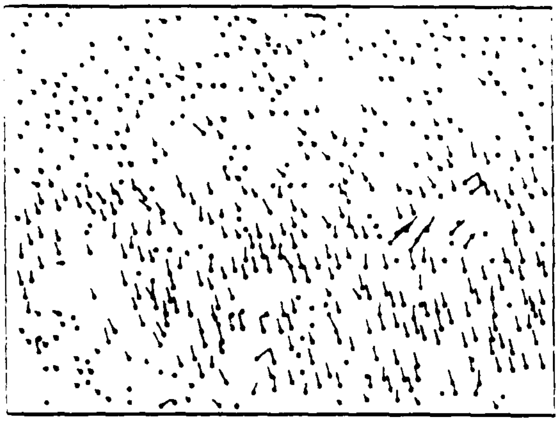


Figure 5: Translational flow field determined from the optical flow field of figure 4.

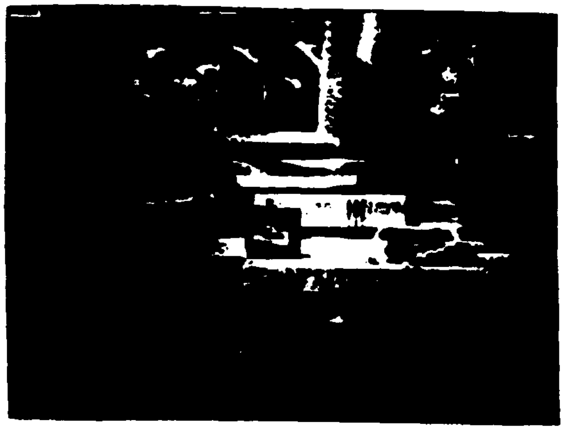


Figure 6: Boundary of a moving object overlaid onto the first image of figure 3.

An image sequence starting with the frame shown in figure 7 is used to illustrate the technique for detecting objects moving in a smooth environment. In this example, the camera moves with respect to an environment consisting of nuts and bolts lying on a planar surface. The optical flow field shown in figure 8 was obtained in the same manner as in figure 4. Figure 9 shows the locations of large variations in optical flow values, corresponding to the boundary of a moving object.

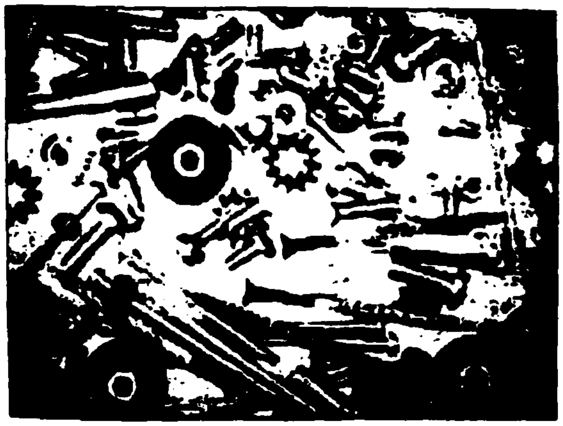


Figure 7: First frame, nuts and bolts sequence.

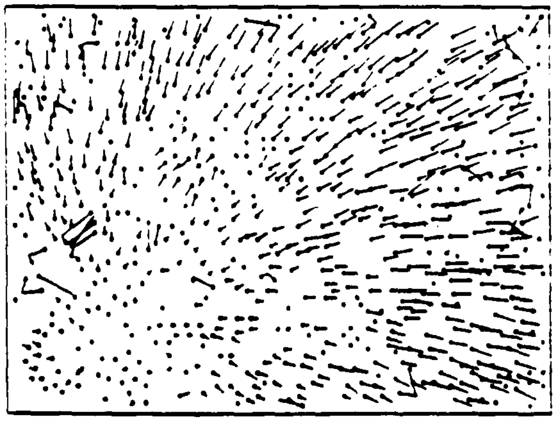


Figure 8: Optical flow field obtained from the image sequence of figure 7.

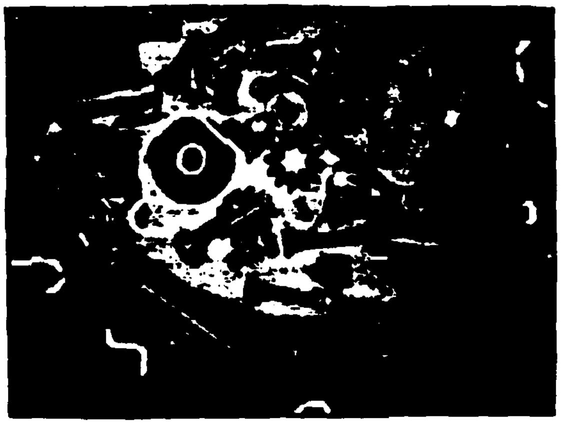


Figure 9: Boundary of a moving object overlaid onto the first image of figure 7.

In figure 10, the circular object in the center of the image is being tracked by the camera while the camera is translating to the right with respect to the environment. Figure 11 shows the estimated optical flow. Figure 12 shows a histogram of the directions of the optical flow. Note that there are two distinct peaks in the histogram. The highest peak corresponds to the optical flow vectors associated with the background and the second peak corresponds to the optical flow vectors associated with the box and the table in the foreground. The variation in flow direction over the image was computed to be approximately 26°, indicating that the tracked object was in fact moving.

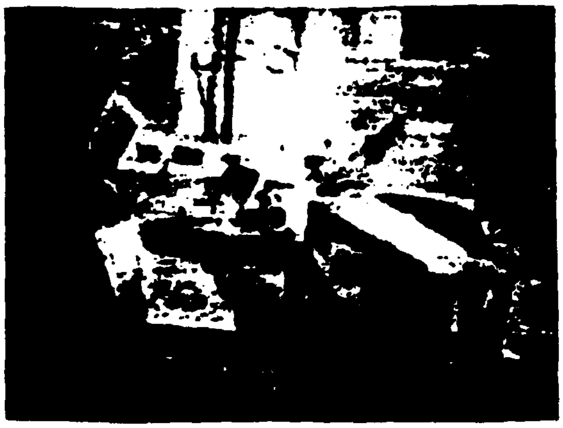


Figure 10: First frame of second indoor scene.

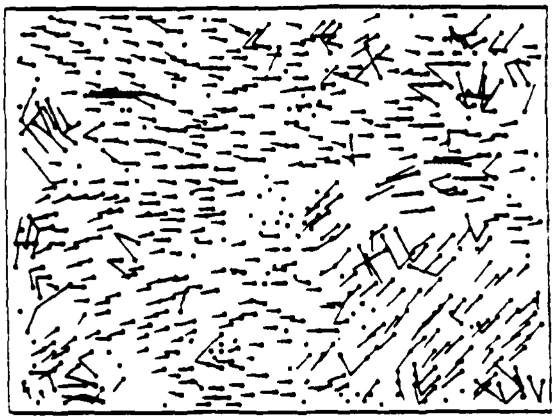


Figure 11: Optical flow field obtained from the image sequence of figure 10.

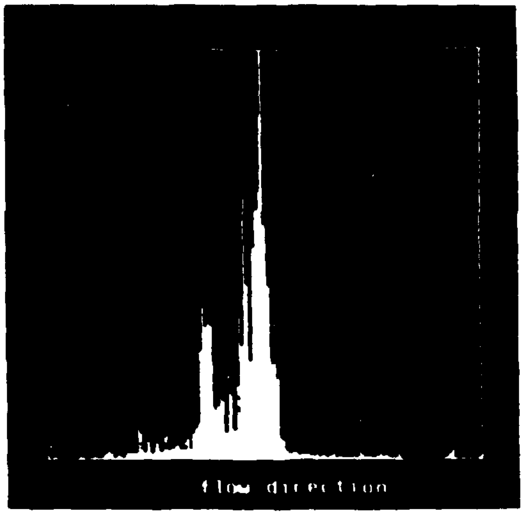


Figure 12: Histogram of the flow directions of the optical flow vectors in figure 10.

As a comparison, a similar experiment in which the tracked object is stationary with respect to the environment while the camera is moving was also preformed. A pair of images similar to that of figure 10 were obtained. The resulting estimated optical flow field is shown in Figure 13. Its corresponding histogram is shown in figure 14. Note that only one distinct peak is observed in this histogram. The global variation in flow direction in this case was computed to be approximately 14° which is significantly smaller than that of the previous example.

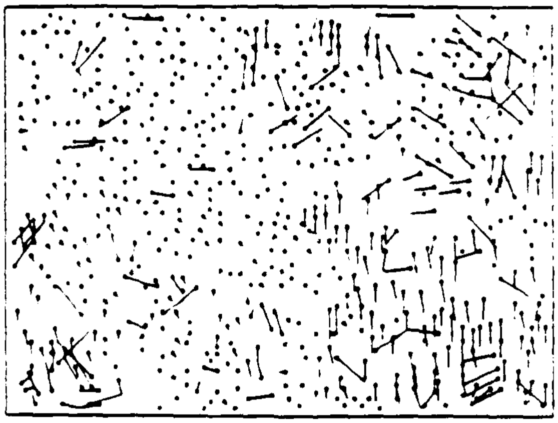


Figure 13: Optical flow field obtained from tracking an object which is stationary with respect to the environment.

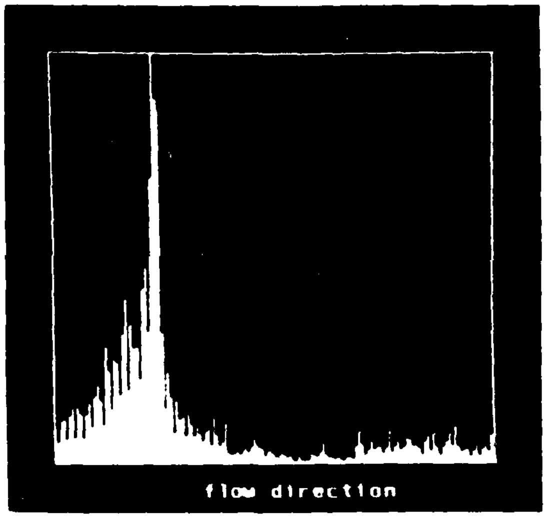


Figure 14: Histogram of the flow directions of the optical flow vectors in figure 13.

5. Discussion.

The methods described above can be grouped into three classes. *Point-based* techniques (known motion, known translation) compare individual optical flow vectors against some standard to determine incompatibilities with the motion of the camera relative to the environment. In all cases described here, the compatibility measure is based on a directional constraint associated with the focus of expansion of the translational flow field. Point-based methods have the advantages of computational simplicity and the ability to detect very small moving objects. They will be most effective when parameters of motion are known precisely and the magnitude of the translational flow field at the point in question is sufficiently large to allow an accurate estimate of direction. *Edge-based* techniques (known rotation, smooth surface) roughly correspond to traditional edge detection. Edge-based motion detection is characterized by the differential flow properties examined and by the filtering technique used to separate edges due to range discontinuities from those due to moving objects. The approach is effective when surfaces are smooth and techniques exist for accurately locating those range discontinuities that do exist. Edge-based methods have the advantage of specifying the outline of moving objects that are detected. They are likely to be of limited use when moving objects are quite small. *Region-based* techniques (tracked object) examine optical flow values over a region, searching for distributions incompatible with rigid motion. As with edge-based approaches, the viewed region must include portions of both object and environment. As long as the region includes portions of both object and environment, this is an effective test for moving objects that does not require any information about camera motion. The region-based method based on tracking potentially moving objects does not require any information about camera motion, but does require that there be significant variations in range over the visible portions of the environment.

One region-based technique not discussed above is based on an explicit check for rigidity. Several structure-from-motion algorithms provide an estimate of rigidity [11,12,13]. Such checks can presumably be used to recognize non-rigid motion due to the presence of a moving object. Numerical structure-from-motion algorithms have proven to be unsatisfactory in practice due to severe problems with ill-conditioning. It is not yet clear whether or not the test for rigidity can be performed in a sufficiently noise tolerant manner to provide for reliable moving object detection.

No method for detecting moving objects will be effective if it depends on knowing precise values of optical flow. Techniques for estimating optical flow are intrinsically noisy (e.g. see [14]). Additional difficulties arise due to the idealized nature of equations (1) - (3). Real cameras are not point projection systems. Substantial effort is required to accurately determine the values of x and y in (2) and (3). Geometric distortions in the optical and sensing systems affect measured locations on the image plane. Variabilities in effective focal length to to focus can be substantial. Reliable techniques will be based on searching for large magnitude effects in the flow field [15]. All of the methods described above compare flow vectors to some predetermined standard, or look for significant differences across flow boundaries. As a result, all deal with relatively large magnitude effects, though reliability is dependent on scene structure, the nature of camera motion, and position in the visual field relative to the direction of translation.

Many of the techniques described above are based on comparing flow values at different points within the field of view. All of these methods require that measurable optical flow exist for points both in the environment and on moving objects. (Some require only that the translational flow be measurable.) Such methods share three important limitations: 1) they are ineffectual near the FOE, 2) the camera must be moving, and 3) portions of the visible environment must be sufficiently close to generate recognizably non-zero translational flow values. Near the FOE, flow due to the environment will be close to zero, regardless of range. If the camera is not moving, all environmental flow values will be zero. The same is true if all points in the environment are very distant relative to the speed of translation. These limitations do not apply just to the methods listed above, as illustrated by figure 1, they are general problems associated with any vision-based motion detection scheme that does not have accurate information about camera translation and/or range to visible surface points.

BIBLIOGRAPHY

[1] T.J. Broida and R. Chellappa, "Estimation of object motion parameters from noisy images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, January 1986.

[2] A.R. Bruss and B.K.P. Horn, "Passive Navigation," *Computer Vision, Graphics, and Image Processing*, v. 21, n. 1, pp. 3-20, 1983.

[3] R.C. Jain, "Segmentation of Frame Sequences Obtained by a Moving Observer," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. PAMI-6, no. 5, pp. 624-629, September 1984.

[4] G Adiv, "Inherent ambiguities in recovering 3-D motion and structure from a noisy flow field," *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, June 1985.

[5] D.H. Ballard and O.A. Kimball, "Rigid body motion from depth and optical flow," *Computer Vision, Graphics, and Image Processing*, vol. 21, pp. 3-20, 1983.

[6] D.A. Marr, *Vision*, San Francisco: W.H. Freeman and Company, 1982.

[7] A.M. Waxman and J.H. Duncan, "Binocular image flows," *Proc. Workshop on Motion: Representation and Analysis*, pp. 31-38, May 1986.

[8] T.S. Huang, S.D. Blostein, A. Werkheiser, M. McDonnel, and M. Lew, "Motion detection and estimation from stereo image sequences: Some preliminary experimental results," *Proceedings Workshop on Motion: Representation and Analysis*, May 1986.

[9] A. Bandopadhay, B. Chandra, and D.H. Ballard, "Active navigation: Tracking an environmental point considered beneficial," *Proceedings Workshop on Motion: Representation and Analysis*, May 1986.

[10] S.T. Barnard and W.B. Thompson, "Disparity analysis of images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. PAMI-2, pp. 333-340, July 1980.

[11] S. Ullman, "Maximizing rigidity: The incremental recovery of 3-D structure from rigid and rubbery motion," *Perception*, 1984.

[12] A. Mitiche, S. Seida, and J.K. Aggarwal, "Determining position and displacement in space from images," *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, June 1985.

[13] E.C. Hildreth and N.M. Grzywacz, "The incremental recovery of structure from motion: Position vs. velocity based formulations," *Proceedings Workshop on Motion: Representation and Analysis*, May 1986.

[14] J.K. Kearney and W.B. Thompson, "Gradient based estimation of disparity," *Proc. IEEE Conf. on Pattern Recognition and Image Processing*, June, 1982.

[15] "Inexact vision," *Proceedings Workshop on Motion: Representation and Analysis*, May 1986.

·ABSTRACTS OF THE AAAI WORKSHOP ON PHYSICAL AND BIOLOGICAL APPROACHES TO COMPUTATIONAL VISION

THE $2\frac{3}{4}$-D SKETCH

William B. Thompson *Lincoln G. Craton* *Albert Yonas*

Computer Science Department Institute of Child Development Institute of Child Development

University of Minnesota
Minneapolis, MN 55455

1 Introduction.

It has been known for many years that motion information provides a cue for depth. Two rather distinct types of information are provided. Relative motion of surface points is an indication of the relative depth of the points. (In this article, we will use the term *depth* to indicate the range from the observer to visible surface points.) If the surface points in question are part of the same rigid object, the analysis of relative visual motion leads to the structure-from-motion and motion-from-structure algorithms currently receiving much attention. Motion parallax also generates relative visual motion that provides information about the overall spatial layout of a scene. The second motion cue to depth occurs at dynamic occlusion boundaries. Surfaces on either side of such boundaries are moving visual with respect to one another. Until recently, it was though that the depth cue at dynamic occlusion boundaries was due to the appearance (*accretion*) or disappearance (*deletion*) of surface texture due to the occlud*ed* surface being progressively uncovered or covered by the occlud*ing* surface.

We have shown that there is an alternate source of information for relative depth at dynamic occlusion boundaries. This information comes from the relative motion of the boundary itself with respect to the surfaces on either side. The investigation of this new cue to depth at surface boundaries is an excellent example of the productive interaction between research in computational models of vision and research in perceptual psychophysics. We start by outlining the computational theory of determining depth at boundaries due to motion. Next, we describe experiments designed to determine whether this cue is used in human perception. We finish with a number of open questions raised by this research. In particular, we argue that Marr's $2\frac{1}{2}$-D sketch is inadequate for representing surface boundaries.

2 The Boundary Flow Constraint.

Visual motion can be used to locate surface boundaries [1]. Edges in an image due to motion can arise from far fewer causes than static image cues such as brightness, color, and texture. In particular, a discontinuity in optical flow can occur *only* because there is a corresponding discontinuity in depth and/or two separate objects are moving with respect to one another. Perhaps even more important, motion provides information

This work was supported by AFOSR contract AFOSR-87-0168, NSF Grant DCR-8500899, and NICHD Grant HD-16924.

and Barrow and Tennenbaum [6]. Marr and Barrow and Tennenbaum suggest a computational architecture with a bottom-up, linear data flow. Use of the boundary flow constraint requires that the boundary be found. the motion of the boundary determined, and the motion of the surrounding surfaces be determined prior to the determination of relative depth. To complicate the computation further, the boundary itself may be signaled only by visual motion. The linear data flow model imposes a predefined ordering on computational operations. It is not clear what ordering could work for boundary flow analysis and still perform adequately for the many other types of low-level computations that are required.

There is an even more important implication. Marr's $2\frac{1}{2}$-D sketch was proposed, in part, as an alternative to the purely 2-D segmentation-based representations that were then popular. The $2\frac{1}{2}$-D sketch was considered as an advantage as it provided 3-D information about surfaces, while not requiring the global organization of the image into "objects". The $2\frac{1}{2}$-D sketch shares one critical deficiency with segmentation-based representations, however. Both are two-dimensional representational structures. Edges in these representations are separations between two regions differing in some visual property. What is missing is any indication of the asymmetric nature of boundaries: edges corresponding to surface boundaries provide information about the occluding surface, but not the occluded surface. Thus, we need something like a $2\frac{3}{4}$-D sketch in which overlapping surfaces can be described.

One explanation of why the subjective contour displays are more effective than the objective contour displays is that the particular subjective contour that was used is a less ambiguous indicator of a depth discontinuity than is the simple straight line which could have arisen from many different causes. The suggestion is that some image cues suggest the existence of an "unsigned" depth boundary [7]. This cues indicate that one surface is in front of another, without indicating which of the surfaces is actually nearer. Cues such as boundary flow can then be used to determine that sign of the depth change. Computational analysis of this sort requires a representation of boundaries more sophisticated than that provided by current models.

References

[1] W.B. Thompson, K.M. Mutch, and V.A. Berzins, "Dynamic occlusion analysis in optical flow fields," *IEEE Trans. Pattern Analysis and Machine Intelligence*, July 1985.

[2] G.A. Kaplan, "Kinetic disruption of optical texture: The perception of depth at an edge," *Perception and Psychophysics*, vol. 6, pp. 193-198, 1969.

[3] A. Yonas, L.G. Craton, and W.B. Thompson, "Relative motion: Kinetic information for the order of depth at an edge," *Perception & Psychophysics*, vol. 41, no. 1, 1987.

[4] L.G. Craton and A. Yonas, "Infants' sensitivity to relative motion information for depth at an edge," submitted to *Child Development*.

[5] D.A. Marr, *Vision*, San Francisco: W.H. Freeman and Company, 1982.

[6] H.G. Barrow and J.M. Tennenbaum, "Recovering intrinsic scene characteristics from images," in *Computer Vision Systems*. A.R. Hanson and E.M. Riseman, eds., New York: Academic Press, 1978.

[7] J.M. Farber and A.B. McConkie, "Optical motions as information for unsigned depth," *Journal of Experimental Psychology: Human Perception and Performance*, vol. 5, no. 3, 1979.

[illegible]

[illegible]

OCCLUSION-SENSITIVE MATCHING

William B. Thompson   Rand P. Whillock

Computer Science Department
University of Minnesota
Minneapolis, MN 55455

Abstract

Model-based recognition of partially occluded objects is a difficult task because of the need to accept matches in which only a subset of model features correspond to image features. Most approaches to implementing these partial matches are subject to serious problems due to ambiguity. Improvements in performance are possible by directly exploiting evidence for occlusion in the image. Once a potential match has been hypothesized, occlusion cues can be used to predict portions of an object model that are not likely to be visible in the image. We describe both an algorithm for matching using occlusion cues, and a method for determining the presence of occlusion based only on image properties. Occluding surfaces are recognized with an approach that combines motion and contrast information. The method accurately localizes edges, detects only those edges likely to correspond to surface boundaries, and provides an indication of which side of an edge corresponds to the occluding surface.

1 Introduction.

Many computational models for object recognition depend in some way on matching two-dimensional object models to image features. 2-D matching is not limited to template matching algorithms. Recently, many recognition approaches have been developed which use three-dimensional part/object models and sophisticated 3-D matching strategies. Because of the highly ambiguous nature of the problem, the final stage in such methods is typically a verification step in which hypothesized information about identification, position, and orientation is used to project a model back into the image to be matched against the actual image features.

Two significant problems plague matching operations. First of all, image features (lines, corners, holes, etc.) cannot be determined in a highly reliable manner. Model features are often missing in the image. Many patterns detected as image features either do not correspond to actual object properties or are not contained within the models. Secondly, in complex scenes objects are often partially occluded. Dealing with occlusion by accepting partial matches increases computational complexity while reducing the reliability of the matching process.

This paper outlines two methods for improving the reliability of matching in the presence of partial occlusion. First, we describe a technique in which visual motion can be combined with static edge cues to improve the effectiveness of the edge detection process. Our technique recognizes that static and dynamic edge cues provide different sorts of information about a boundary. Static cues such as contrast edges give good spatial localization, but are subject to highly ambiguous interpretations. Visual motion is a robust indicator of surface boundaries, but does not yield precise information on the location of the boundary. The approach given here accurately locates edges due to surface boundaries, without generating many "false" edges. Even more importantly, the method gives a direct indication of which side of an edge corresponds to the occluding surface generating the edge.

The second technique uses information about occlusion to aid in the matching process. Most existing matching algorithms that are tolerant of occlusion look for a partial correspondence between model and *image features. If a partial match is found,* unmatched model components are assumed to be hidden by an occlusion. This approach leads to difficulties because of the chances for partial matches occurring coincidentally. In our method, information about occlusion boundaries is used to explicitly identify model features that will not be visible in the image. Most of the remaining model features should be findable if the match is in fact correct. Occluded model features are determined based directly on image properties at boundaries, rather than just on the absence of an image feature at some expected location. The result is a significant decrease in ambiguity.

2 Background.

2.1 Combining motion and contrast information for edge detection.

Segmentation schemes which combine motion and contrast information date back to at least to the work of Jain, Martin, and Aggarwal [1]. This approach used a difference operator between two frames to find areas in the image that had changed due to motion. A static segmenter was then run within these areas to find the boundaries of the moving regions. Thompson used a region merger approach that grouped pixels into regions based on similarities in contrast and motion information [2]. Hayes and Jain developed an edge detector based on a product of the spatial gradient and a temporal operator [3]. The purpose was to

This work was supported by AFOSR contract AFOSR-87-0148 and NSF grant DCR-8500899

limit sensitivity to areas signaled by both static and dynamic effects. More recently, Gamble and Poggio have developed a Markov Random Field model for recovering optical flow in a manner that integrated contrast boundaries with visual motion [4]. Their approach constrained discontinuities in flow to occur only at intensity edges.

Relatively little work has been done on differentiating between occluding and occluded surfaces without resort to fitting object or part models. Waltz used constraints associated with line drawing vertices to identify extremal contours and to determine which side of such a contour corresponded to an occluding surface [5]. Smitley and Bajcsy identified occluding surfaces in stereo imagery by comparing correlations between frames for images patches on either side of a boundary [6]. If the correlations differed substantially, the boundary was assumed to be due to occlusion and the region with the highest correlation between views was assumed to correspond to the occluding surface. Thompson, Mutch, and Berzins showed how edges in optical flow could be used to recognize occluding surfaces [7]. Their approach is discussed in more detail in section 3.

2.2 Matching.

Template matching was one of the first methods proposed for the visual recognition of objects. Template matching utilizes a correlation measure between one or more model patterns and images to be analyzed. Invariance to translation and/or rotation can be obtained by appropriate scanning of the template pattern over an image. While useful in some applications, template matching suffers from problems due to computational complexity and is unable to deal effectively with the matching of three-dimensional models to two-dimensional imagery.

Recognition of three-dimensional objects is often done by using configurations of image features to estimate how a three-dimensional object is being projected into the two-dimensional image. The object model is subjected to the appropriate projection, resulting in a prediction of the objects appearance in the image. A verification process is used to determine if the predicted configuration of object features actually appears in the image (e.g., [8,9,10]). Such methods avoid many of the problems associated with straightforward template matching.

Recognition of partially occluded objects has been a major challenge for many years. Most approaches attempt to find good partial matches between subsets of object models and image features (e.g., [11,12,13]). Allowing for partial matches increases the likelihood of false positive classification errors. In addition, the extraneous configurations of boundaries generated by overlapping objects causes additional confusion.

Some preliminary attempts have been made to directly incorporate occlusion information into the matching process. Fisher developed evidence for extraneous or missing image features based on boundary topology and other information about the depth ordering of surfaces [14]. Specialized heuristics were used to discount the irrelevant mismatches during a verification stage. Castan used the results of a partial matching process to determine estimates of model features likely to be hidden by occlusions [15]. Evidence for visibility and occlusion came from a presumption that visible features were spatially adjacent, rather than from any three-dimensional analysis of the imagery.

3 Motion-based Segmentation.

Thompson, Mutch, and Berzins develop an edge detector for optical flow fields [7]. One important aspect of this work is that motion-based edge detection directly yields information about which side of the edge corresponds to the occluding surface. This identification is based on a comparison between the optical flow on either side of the boundary and the visual motion of the boundary itself. (Aperture effects usually require that all image flows be projected onto an axis parallel to the normal to the edge.) The principle underlying the identification of occluded surfaces is summarized in the *boundary flow constraint:*

> *At a surface boundary, the visual motion of the boundary itself is the same as the visual motion of the surface generating the boundary.*

At a boundary, we need only look at the image-plane motion of the boundary (the *boundary flow*) and the optical flow immediately to either side. Optical flow inconsistent with the boundary flow corresponds to an occluded surface.

One problem with exploiting the boundary flow constraint is the apparent need to determine the actual motion of the boundary. In many circumstances, this can result in a difficult correspondence problem. [7] demonstrated how the motion of optical flow edges can be related to the boundary flow constraint in a manner that does not explicitly compute boundary motion. In that work, the boundary that was moving was itself indicated by a motion cue. Here, we extend the result to show how *any* zero-crossing style edge operator can be easily used to distinguish between occluding and occluded surface. As shown in [7], with an appropriate change of coordinate systems it is sufficient to consider only two cases. In one, two surfaces are moving towards one another with equal but opposite optical flows. In the second case, the surfaces are moving away from one another with equal but opposite flows. Over time, the Laplacian pattern at the boundary will move with the surface to which it is attached. If a zero-crossing edge detector is applied to an optical flow pattern, all that is necessary to classify the edge is to observe the sign of the Laplacian pattern as it translates.

The situation is somewhat more complicated if edges are signaled by some feature other than optical flow. In such cases, it is necessary to consider both the contrast orientation of the edge and the pattern of motion to either side. The sign of the Laplacian function can be used to determine the direction of boundary movement relative to the direction of the gradient at the boundary. If we observe the value of the Laplacian at the zero crossing and that value goes negative, then we know that the edge has moved in the direction of the gradient. If the value of the Laplacian goes positive, then the edge motion is in the direction opposite to the gradient. It is still necessary to compare edge motions and surface motions. Again using the coordinate system transform, we need only determine whether the two surfaces are moving towards or away from each other. It is not necessary to quantitatively estimate actual surface and boundary flows.

The following algorithm implements this process:

1. Find an edge point, $\vec{x}_0$, in frame t_0. Compute the gradient $\nabla I(\vec{x}_0)$, where $I(\vec{x})$ is any perceivable function of $\vec{x}$ that corresponds to surface properties.

2. Project all optical flow values onto an axis parallel to $\nabla I(\vec{x}_0)$.

3. Normalize coordinates by locating an evaluation point $\vec{x}_1 = \vec{x}_0 + \vec{f}_1$ in frame t_1, where $\vec{f}_1$ is the average inter-frame flow in the neighborhood of $\vec{x}_0$.

4a. The direction of $\nabla I(\vec{x}_0)$ points towards the side of the boundary corresponding to the occluding surface if $\nabla^2 G\, I(\vec{x}_1)$ is negative and the two surfaces are approaching one another or if $\nabla^2 G\, I(\vec{x}_1)$ is positive and the two surfaces are separating.

4b. The direction of $\nabla I(\vec{x}_0)$ points towards the side of the boundary opposite the occluding surface if $\nabla^2 G\, I(\vec{x}_1)$ is positive and the two surfaces are approaching one another or if $\nabla^2 G\, I(\vec{x}_1)$ is negative and the two surfaces are separating.

Note that if surface motion is parallel to the boundary, no determination of occluding and occluded surfaces is made. In fact, in this situation no definitive determination is possible based only on visual motion.

One advantage of this particular algorithm is that it directly provides a mechanism for combining motion-based boundary detection with static edge cues. Discontinuities in optical flow can only occur due to discontinuities in depth and/or due to two surfaces moving relative to one another. Thus, flow edges can arise from far fewer causes than edges due to changes in intensity, texture, color, etc. Unfortunately, flow edges are difficult to localize precisely. The above algorithm can be used to filter out all static edges that are not associated with a change in optical flow over the neighborhood of the edge. The effect is to use motion to reduce ambiguity, while using the static cues to preserve localization. In our current algorithm, we are only interested in boundary points at which we can differentiate between occluding and occluded surfaces. As a result, we delete all edge elements that do not have some differential optical flow along an axis perpendicular to the edge. This is easily done by modifying the above algorithm as follows:

3b. If the magnitude of $\nabla^2 G\, I(\vec{x}_1)$ is close to zero, delete the edge element at x_0 from further consideration.

Only a bit more complexity is required in order to recognize edges with differential motion only tangential to the edge orientation. Such edges signal surface boundaries, but it is not possible to distinguish between the occluding and occluded sides.

4 Occlusion-Sensitive Matching.

We have developed a simple model of how occlusion information might be used to aid in recognition. The model uses occlusion cues arising from the boundary flow constraint to reduce ambiguity in template matching applied to partially occluded objects. In presenting this model, our aim is to demonstrate the utility of incorporating occlusion information directly into the recognition process. The specifics of the algorithm are for purposes of illustration only. The approach will work for verification as well as standard template matching. Any occlusion cue can be used: the method is not limited to using just motion information. More efficient and reliable implementations are possible. The basic principles of our approach can be summarized as follows:

- *Determine a matching "score" based on searching for model features in the image.*
- *Introduce penalties for model features not in the image, but only if there is not evidence for the features being hidden by an occlusion.*
- *Do not introduce penalties for image features not accounted for in the model.*

In the examples presented below, we define the matching score to be the percentage of model features found in the image. This is done by computing the ratio of matched model features to potentially matchable model features. The features used in our simple example are silhouette edge elements. Only image edges with differential motion across the edge are used. A small distance toleration is allowed for to accommodate noise and other distortions. Information about occluding edges in the image is used in two ways. First of all, the model/non-model sides of the template edge must be compatible with the occluding/occluded sides of the image edges. (Note that this is a stronger requirement than just orientational compatibility.) Secondly, a model edge element is considered potentially matchable if it is not *masked*. When a model is being matched at a particular image location, masking occurs if there are significant occlusion edges in the image within the interior of the model. Masking regions are "grown" outward from the occluding side of any interior image edges. To assure that it will not extend beyond any occluding surface, the masking region ends at the first image edge reached. In our current implementation, matching is first done without using the masking operation. Areas of partial match are then reevaluated using the masking procedure.

A set of simple examples was created to test our approach of occlusion sensitive matching. We used artificially created objects to better control for ambiguity in matching. However, the examples all involve real imagery and automatically determined optical flow. Figure 1 shows a set of fourteen object models. Two actual objects were used, one **T** shaped, the other **L** shaped. Figure 2 shows one frame from a sequence in which the **T** is moving behind a wall to the right. The wall is partially occluding the **T**. As a result, simple template matching may not be effective for recognition. Figure 3 shows contrast edges in the **T** sequence. The edges were determined using a large kernel zero-crossing operator. Figure 4 shows motion/contrast edges determined by deleting edge

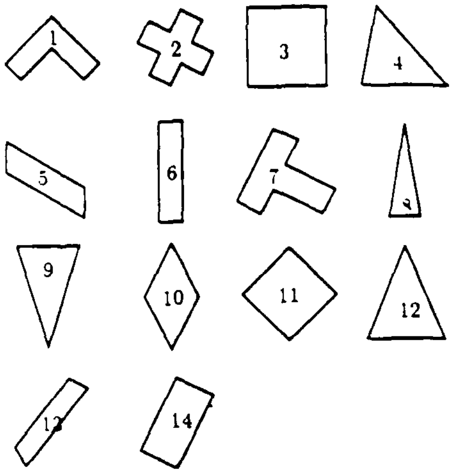


Figure 1: Model set.

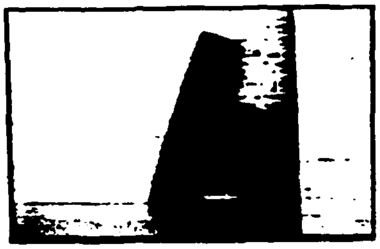


Figure 2: Frame from T image sequence.

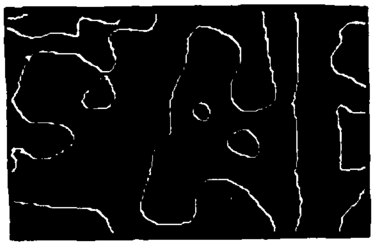


Figure 3: T contrast edges.

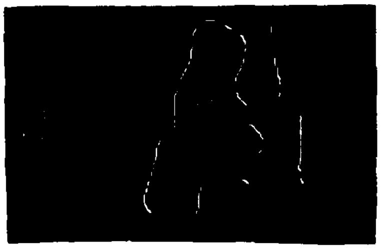


Figure 4: T motion/contrast edges.

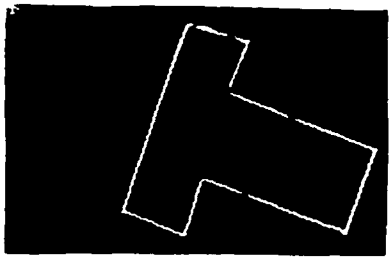


Figure 5: Best fit location for model.

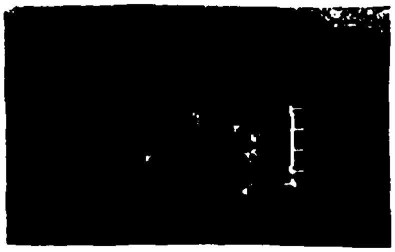


Figure 6: Unmatched edges.

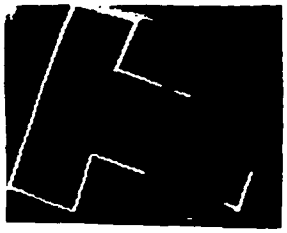


Figure 7: Masked portions of T model.

elements in figure 3 that are not associated with differential optical flow across the edge. Figure 5 shows the position of the T model in the image resulting in the highest matching score. Figure 6 shows the unmatched edges within the T model when applied to the image at the location shown in figure 5. The hash marks along the edges point to the occluding surface, as indicated by the boundary flow constraint. Finally, figure 7 shows the portions of the T model which have been masked as a result of the internal edges shown in figure 6.

Table 1 shows the matching scores for all model types evaluated against the T and L sequences. The highest scores in each column have been italicized. The models are matched against the raw contrast edges, the motion/contrast edges, the motion/contrast edges using the model/non-model orientational compatibility constraint, and finally using all of the matching constraints described above (differential motion, model/non-model edge orientation, and masking). The data, while currently limited to a few test cases, suggests that using occlusion information can reduce ambiguity in matching. Using all of the available matching constraints, both examples are correctly classified. Using either traditional template matching or using only a subset of the matching constraints causes one or both of the images to be misclassified.

5 Summary.

Edge detection is possible based on both contrast and motion information. Contrast edges can arise from a large number of causes and thus are difficult to accurately interpret. Motion edges are always associated with depth and/or surface boundaries, but are difficult to localize precisely. The motion-based segmentation technique described above combines motion and contrast cues in an integrated edge detection process. Localization is based on contrast edges, while motion information is used to filter out edges not likely to correspond to surface boundaries. The method further gives a direct indication of the side of the boundary corresponding to the occluded surface.

Identification of occluded and occluding surface can significantly aid in recognition tasks. We have presented a simple matching algorithm in which the presence of occlusion boundaries is used to avoid penalizing matches for situations in which model features are hidden from view by other objects. While our algorithm has been described within the context of template matching, it is equally appropriate when verifying hypothesized matches suggested by more complex three-dimensional reasoning processes.

References

[1] R. Jain, W.N. Martin, and J.K. Aggarwal. "Segmentation through the detection of changes due to motion". *Computer Graphics and Image Processing*, 11:13–34, 1979.

[2] W.B. Thompson. "Combining motion and contrast for segmentation". *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-2:543–549, November 1980.

[3] S.M Haynes and R. Jain. "Detection of moving edges". *Computer Vision, Graphics and Image Processing*, 21:345–367, March 1983.

[4] E. Gamble and T. Poggio. *Visual Integration and Detection of Discontinuities: The Key Role of Intensity Edges*. AI Memo 970, MIT, 1987.

[5] D. Waltz. "Understanding line drawings of scenes with shadows". In P.H. Winston, editor, *The Psychology of Computer Vision*, McGraw-Hill, New York, 1975.

[6] D.L. Smitley and R. Bajcsy. "Stereo processing of aerial, urban images". *Proc. Seventh Int. Conference on Pattern Recognition*, 433–435, 1984.

[7] W.B. Thompson, K.M. Mutch, and V.A. Berzins. "Dynamic occlusion analysis in optical flow fields". *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-7:374–383, July 1985.

[8] L.G. Roberts. "Machine perception of three-dimensional solids". In J.T. Tippett et al., editors, *Optical and Electro-Optical Information Processing*, MIT Press, Cambridge, MA, 1965.

[9] W.E.L. Grimson. "Recognition of object families using parameterized models". *Proc. First International Conference on Computer Vision*, 93–101, 1987.

[10] D.P. Huttenlocher and S. Ullman. "Object recognition using alignment". *Proc. First International Conference on Computer Vision*, 102–111, 1987.

[11] W.A. Perkins. "Model-based vision system for scenes containing multiple parts". *Proc. Fifth International Joint Conference on Artificial Intelligence*, 678–684, 1977.

[12] J.W. McKee and J.K. Aggarwal. "Computer recognition of partial views of curved objects". *IEEE Trans. on Computers*, C-26:790–800, 1977.

[13] R.C. Bolles. "Robust feature matching through maximal cliques". *Proc. SPIE Technical Symposium on Imaging Applications for Automated Industrial Inspection and Assembly*, 1979.

[14] R. Fisher. "Using surfaces and object models to recognize partially occluded objects". *Proc. Eighth International Joint Conference on Artificial Intelligence*, 989–995, 1983.

[15] S. Castan, J. Shen, and N.Q. He. "A method for recognition and positioning of partially observed objects". *Proc. Eighth Int. Conference on Pattern Recognition*, 1986.

Model	T image sequence				L image sequence			
	contrast edges	motion/ contrast	model/ non-model	occlusion masking	contrast edges	motion/ contrast	model/ non-model	occlusion masking
M_1 (L)	.635	.422	345	.439	*.847*	.594	550	*637*
M_2 (Cross)	.659	562	511	542	.754	517	448	154
M_3 (Square)	.642	248	215	296	.512	.260	192	192
M_4 (Asymmetric triangle)	628	520	.456	603	416	.321	257	277
M_5 (Quadrilateral)	.652	380	295	.526	761	.377	.338	338
M_6 (Rectangle)	*.800*	548	388	638	.704	504	356	356
M_7 (T)	.670	532	494	*667*	543	327	270	286
M_8 (Narrow triangle)	.665	543	520	520	715	498	412	412
M_9 (Inverted triangle)	.769	474	446	475	.713	478	430	430
M_{10} (Narrow diamond)	.621	*571*	*566*	606	797	*648*	571	571
M_{11} (Standard diamond)	583	456	406	437	.772	594	556	559
M_{12} (Broad triangle)	563	540	.398	.525	.716	425	372	380
M_{13} (Tilted trapezoid)	.635	.450	375	551	745	625	*590*	590
M_{14} (Tilted rectangle)	574	426	413	439	702	603	554	561

Table 1: Matching scores – all models applied to T and L sequences.

Structure-From-Motion By Tracking Occlusion Boundaries

William B. Thompson

Computer Science Department
University of Minnesota
Minneapolis, MN 55455

To appear in the *Proceedings of the IEEE Workshop on Visual Motion*, 1989.

Abstract

Active visual tracking of points on occlusion boundaries can simplify certain computations involved in determining scene structure and dynamics based on visual motion. Two such techniques are described here. The first provides a measure of ordinal depth by distinguishing between occluding and occluded surfaces at a surface boundary. The second can be used to determine the direction of observer motion through a scene.

1 Introduction.

The study of computational models of *active vision* has received a flurry of recent activity (e.g., [1,2,3]). These and similar papers have investigated ways in which the visual process can be simplified and/or extended if active control is available over camera motion. Much of this work has dealt specifically with the issue of eye/camera rotation [2,3]. The ability to visually track environmental points can lead to significant simplifications in computing visual properties. This note describes two such simplifications, both involving the tracking of edge points corresponding to occlusion boundaries. The first technique determines local depth orderings by recognizing which side of a boundary corresponds to an occluding surface. The second technique is able to estimate the direction of observer motion in a simpler manner than most other, previously proposed approaches.

The methods described below are most effective when the following three assumptions hold: An observer is moving though an environment in which at most a relatively small portion of the visual

This work was supported by AFOSR contract AFOSR-87-0168 and NSF Grants DCR-8500899 and IRI-8722576.

field corresponds to moving objects. Occlusion boundaries involving significant changes in depth commonly occur. The observer is able to keep a selected edge element centered in the field of view. This last assumption is at least plausible in most natural situations where boundaries are not straight and/or surfaces are visually textured. Analysis will be based on optical flow in the image near the tracked edge element. Note that in biological terms, this corresponds to retinal flow, not the Gibsonian idea of flow in the "optic array".

2 Analysis.

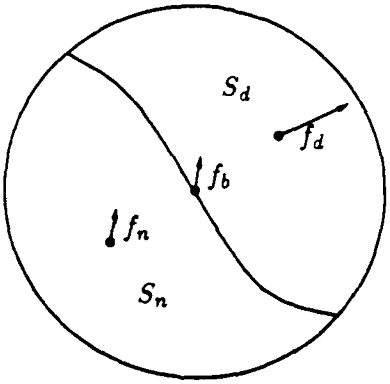


Figure 1: Optical flow near a surface boundary.

Visual motion depends on the instantaneous translational velocity of the eye/camera, the range to surface points in the scene, and the rotational velocity needed to track a particular scene point. Figure 1 illustrates the situation in the neighborhood of a boundary when no rotation is occurring. S_n corresponds to a near surface, which has associated optical flow f_n. S_n is occluding a more distant surface S_d, with associated flow f_d. The boundary itself moves in the image with flow f_b. From [4], we know that close to an occlusion boundary the visual motion of the occluding surface and the visual motion of the boundary are the same. Thus, $f_b = f_n$. Figure 2 describes

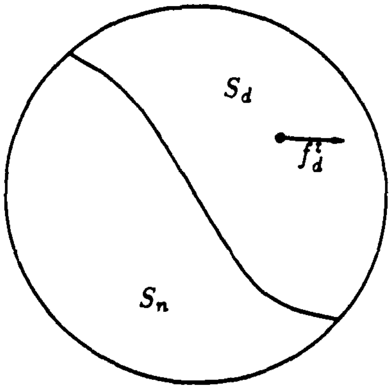


Figure 2: Optical flow with edge tracking.

the situation when the edge is being accurately tracked. Tracking is effected by introducing an eye/camera rotation of velocity $\omega = (A, B, 0)^T$ which exactly compensates for f_b. This also has the effect of nulling out f_n. The only visible flow left, $f_d^t = f_d - f_b$, is associated with the more distant surface.

A simple set of equations defines the relationship between optical flow, motion, and scene structure [5]. Using a planar imaging system, perspective projection, and a coordinate system centered at the camera with z axis along the line of sight:

$$u = u_t + u_r \quad , \quad v = v_t + v_r \tag{1}$$

where u and v are the x and y components of flow, z is the distance to the surface point imaged at (x, y), translational velocity is $\mathbf{T} = (U, V, W)^T$, and

$$u_t = \frac{-U + xW}{z} \quad , \quad v_t = \frac{-V + yW}{z} \tag{2}$$

$$u_r = Axy - B(x^2 + 1) \quad , \quad v_r = A(y^2 + 1) - Bxy \tag{3}$$

The optical flow equations simplify considerably at the center of the field of view:

$$\lim_{x,y \to 0} u_t = \frac{-U}{z} \quad , \quad \lim_{x,y \to 0} v_t = \frac{-V}{z} \tag{4}$$

$$\lim_{x,y \to 0} u_r = -B \quad , \quad \lim_{x,y \to 0} v_r = A \tag{5}$$

If the tracked boundary element is centered within the field of view and if surface flow is measured near this center, then f_b, f_n, and f_d are all determined by equations 4–5.

Utilizing the fact that $z_n < z_d$, we can now compute f_d^t:

$$f_d^t = f_d - f_b = f_d - f_n \tag{6}$$

$$= \left(\frac{-U}{z_d} - B - \frac{-U}{z_n} + B \, , \, \frac{-V}{z_d} + A - \frac{-V}{z_n} - A\right) \tag{7}$$

$$= \left(\left(\frac{1}{z_n} - \frac{1}{z_d}\right)U \, , \, \left(\frac{1}{z_n} - \frac{1}{z_d}\right)V\right) \tag{8}$$

$$= (aU, aV) \quad , \quad a > 0 \tag{9}$$

f_d^t is thus a scaled version of the projection of the translation vector onto the image plane.

We can now summarize the two algorithms for analyzing visual motion using edge tracking:

- *Identification of occluding surface.*

  When a boundary element is visually tracked, the region to the side of the boundary corresponding to the occluding surface will have near-zero image flow. The region to the side of the boundary corresponding to the occluded surface will in general be associated with significant visual motion.

- *Determination of direction of observer motion.*

  When a boundary element is visually tracked. optical flow due to the more distant surface indicates the direction of observer motion. The flow vectors point in the direction of the image location corresponding to the line of sight coincident with the direction of translational motion. (This location is commonly called the "focus of expansion", but the term is only strictly correct for purely translational motion.) Multiple fixations over the field of view can be used to solve for the actual direction of translation.

3 Discussion.

Both algorithms offer significant computational simplifications over alternate approaches. The few previously reported optical flow based techniques for differentiating between occluding and occluded surfaces require reasonably accurate flow estimates on either side of the boundary [4,6]. The method reported here only requires that regions of significant image motion be recognized. It is far easier to determine that image motion is occurring than it is to estimate the specific characteristics of that motion. When eye/camera rotations are possible, the determination of observer motion is difficult because of the complex manner in which translational and rotational motion interact to generate an optical flow field (see [5]). Edge tracking eliminates the complexity associated with rotation.

It is important to note that eye tracking does not reduce the *conceptual* difficulties associated with these two tasks. Eye tracking provides neither additional constraints nor other sorts of new information. This is easily seen by recognizing that all of the information in the tracking image is available in an image of the same scene without tracking. Tracking is accomplished by generating a rotation of the eye/camera system based on estimates of image drift such as optical flow at the image center. Once this rotational velocity is determined, a non-tracking image sequence can trivially be converted into the equivalent tracking sequence using equation 3. In fact, both of the algorithms described above are really special cases of methods already presented in the literature. Occlusion analysis is described in [4]. The method for determining direction of motion is essentially equivalent to that described in [7]. What is different are the simplifications in actual algorithms, not the underlying computational theory.

The effectiveness of these two algorithms is limited by the accuracy with which boundaries can be tracked and by the visual texture present adjacent to the boundaries. While biological systems are capable of tracking environmental points with relatively high precision, the computer vision community has only recently begun to study the engineering difficulties involved in tracking features in complex scenes. Aperture effects are a further consideration. It is generally felt that only the component of motion perpendicular to an edge can be determined. This is actually only true if the edge does not curve (e.g., see [8]). Reasonably reliable two-dimensional tracking should be possible for most realistic scenes, though sufficient experimentation has not yet been done. Both algorithms depend on recognizing aspects of image motion in the neighborhood of the tracked edge. This is most easily accomplished if both surfaces are visually textured. This will hold in many but not all scenes. We do know that human vision is capable of "filling in" the motion of homogeneous portions of surfaces. We do not as yet have good computational models of how this is done. however.

Open questions remain as to whether or not biological vision systems actually use methods of this sort to simplify the determination of scene structure and motion trajectories. To answer these questions. we need to know more about fixation patterns in realistic dynamic environments and about how fixation and eye tracking affect the perception of relative depth.

Acknowledgement.

Dana Ballard first got me thinking about the role of tracking in visual motion analysis.

References

[1] R. Bajcsy, "Active perception vs. passive perception", *Proc. IEEE Workshop on Computer Vision*, 55–59, 1985.

[2] J. Aloimonos, I. Weiss, and A. Bandyopadhyay, "Active vision", *Proc. First International Conference on Computer Vision*, 35–54, 1987.

[3] D.H. Ballard, *Eye Movements and Spatial Cognition*, Technical Report 218, University of Rochester, 1987.

[4] W.B. Thompson, K.M. Mutch, and V.A. Berzins, "Dynamic occlusion analysis in optical flow fields", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-7:374–383, July 1985.

[5] B.K.P. Horn, *Robot Vision*, MIT Press, Cambridge, MA, 1986.

[6] W.F. Clocksin, "Perception of surface slant and edge labels from optical flow: a computational approach", *Perception*, 9:253–269, 1980.

[7] J.H. Reiger and D.T. Lawton, "Sensor motion and relative depth from difference fields of optic flows", *Proc. Eighth International Joint Conference on Artificial Intelligence*, 1027–1031, 1983.

[8] E.C. Hildreth, *The Measurement of Visual Motion*, MIT Press, Cambridge, MA, 1983.